TOMSK POLYTECHNIC UNIVERSITY ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Томский политехнический университет» (ТПУ)

Инженерная школа информационных технологий и робототехники
Направление подготовки 01.03.02 Прикладная математика и информатика
Отделение информационных технологий

БАКАЛАВРСКАЯ РАБОТА

Тема работы
Разработка алгоритма для анализа и обработки больших данных

УДК 004.021-047.84:004.6

Студент

Группа	ФИО	Подпись	Дата
8Б51	Белогуров Артём Игоревич		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Губин Евгений Иванович	Кандидат физико-математических наук		

КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН ШБИП	Подопригора Игнат Валерьевич	к.э.н., доцент		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент ООТД	Мезенцева Ирина Леонидовна			

ДОПУСТИТЬ К ЗАЩИТЕ:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Руководитель ООП	Шевелев Геннадий Ефимович	к.ф-м.н., доцент		

Томск – 2019 г.

TOMSK POLYTECHNIC UNIVERSITY ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Томский политехнический университет» (ТПУ)

Инженерная школа информационных технологий и робототехники
Направление подготовки 01.03.02 Прикладная математика и информатика
Отделение информационных технологий

УТВЕРЖДАЮ:
И.о. руководителя
ОИТ ИШИТР ТПУ
Шерстнев В.С.
_____

ЗАДАНИЕ
на выполнение выпускной квалификационной работы

В форме:

бакалаврской работы

Студенту:

Группа	ФИО
8Б51	Белогуров Артём Игоревич

Тема работы:

Разработка алгоритма для анализа и обработки больших данных	
Утверждена приказом (дата, номер)	№1101/с от 12.02.2019

Срок сдачи студентом выполненной работы:	

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

Исходные данные к работе	- объект исследования: банковские данные взятые из библиотеки SAS; - объектно-ориентированный язык программирования Python; - свободная интегрированная среда разработки приложений Google Colaboratory; - литературные источники
Перечень подлежащих к исследованию, проектированию и разработке вопросов	- разработка алгоритма для предобработки данных; - реализация методики предобработки данных с помощью Python; - сравнение над обработанными данными и необработанными; - сравнительный анализ результатов работы различных реализаций.
Перечень графического материала	- мультимедийная презентация в формате ppt

Консультанты по разделам выпускной квалификационной работы	
Раздел	**Консультант**
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Подопригора Игнат Валерьевич

Социальная ответственность	Мезенцева Ирина Леонидовна

Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику	

Задание выдал руководитель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент ОИТ	Губин Евгений Иванович	-		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8Б51	Белогуров Артём Игоревич		

TOMSK POLYTECHNIC UNIVERSITY ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Томский политехнический университет» (ТПУ)

Инженерная школа информационных технологий и робототехники
Направление подготовки 01.03.02 Прикладная математика и информатика
Отделение информационных технологий

Форма представления работы:

Бакалаврская работа

КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН
выполнения выпускной квалификационной работы

Срок сдачи студентом выполненной работы:	14.06.2019 г.

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
14.06.2019 г.	Основная часть	75
14.06.2019 г.	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	15
11.06.2019 г.	Социальная ответственность	10

Составил преподаватель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент ОИТ	Губин Е.И..	Кандидат физико-математических наук		

СОГЛАСОВАНО:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Руководитель ООП	Шевелев Г.Е.	к.ф-м.н., доцент		

ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ»

Студенту:

Группа	ФИО
8Б51	Белогуров Артём Игоревич

Инженерная школа	Информационных технологий и робототехники	**Отделение**	Информационных технологий
Уровень образования	Бакалавриат	**Направление/специальность**	Прикладная математика и информатика

Разработка алгоритма для анализа и обработки больших данных	
Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:	
1. Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих	Целью данной главы ВКР является разработка алгоритма для анализа и предобработки данных.
2. Нормы и нормативы расходования ресурсов	
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
1. Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения	Оценка потенциального потребления исследования; swot-анализ.
2. Определение возможных альтернатив проведения научных исследований	Определение возможных альтернатив с помощью морфологического подхода.
3. Планирование научно-исследовательских работ	Планирование структуры работ, определение трудоемкости работы и построение календарного графика.
Перечень графического материала:	
1. Календарный план-график	

Дата выдачи задания для раздела по линейному графику	

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Подопригора И.В.	Кандидат экономических наук		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8Б51	Белогуров Артём Игоревич		

ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа	ФИО
8Б51	Белогуров Артём Игоревич

Инженерная школа	Информационных технологий и робототехники	**Отделение**	Информационных технологий
Уровень образования	Бакалавриат	**Направление/специальность**	Прикладная математика и информатика

Разработка алгоритма для анализа и обработки больших данных	
Исходные данные к разделу «Социальная ответственность»:	
1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения.	Объект исследования – программное обеспечение, реализующее анализ и обработку данных, а также их скрининг. Рабочая зона – аудитория с естественным и искусственным освещением, оборудованная системой отопления и кондиционирование воздуха. Область применения – любая организация, работающая с банковскими данными предназначенных для выдачи кредитов.
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
1. Производственная безопасность 1.1 Анализ выявленных вредных факторов проектируемой производственной среды:	Анализ выявленных вредных факторов при разработке и эксплуатации включает: Микроклимат, слабая освещенность, повышенный уровень шума, умственное перенапряжение, монотонность труда. Требования к помещению описаны в СанПиН 2.2.4.3359-16.
1.2 Анализ выявленных опасных факторов проектируемой произведённой среды:	Анализ выявленных опасных факторов при разработке и эксплуатации включает: Удар электрическим током, короткое замыкание, статическое электричество
2. Экологическая безопасность	Утилизация использованной техники (компьютера и других составляющих аппаратно-программного комплекса). Утилизация канцелярских принадлежностей и бумаги, использованных лампочек.
3. Безопасность в чрезвычайных ситуациях:	Возможная чрезвычайная ситуация для данного помещения – пожар, вследствие короткого замыкания. Необходимо установить общие правила Поведения при пожаре, ознакомить с Планом эвакуации, иметь в наличии

	исправного огнетушителя, назначить ответственного по пожарной безопасности в данном помещении.
4. Правовые и организационные вопросы обеспечения безопасности: – специальные (характерные для проектируемой рабочей зоны) правовые нормы трудового законодательства; – организационные мероприятия при компоновке рабочей зоны	Основные проводимые правовые и организационные мероприятия по обеспечению безопасности трудящихся в аудиториях. Анализ правильного расположения и организации рабочего места, а также режима работы. Расположения и организации рабочего Регулирования организации рабочего места при выполнении работы сидя проводятся согласно ГОСТ 12.2.032-78. Трудовые отношения регулируются согласно ТК РФ ФЗ–197 от 30.12.2001.

Дата выдачи задания для раздела по линейному графику	

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент ООТД	Мезенцева И.Л.			

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8Б51	Белогуров Артём Игоревич		

Планируемые результаты обучения по направлению

01.03.02 «Прикладная математика и информатика»

Код результата	Результат обучения
Профессиональные компетенции	
Р1	Применять глубокие математические и профессиональные знания для решения задач научно-исследовательской, проектной, производственной и технологической деятельности в области системного и прикладного программирования.
Р2	Умение использовать знания по естественнонаучным дисциплинам при определении задач математического моделирования объектов и явлений в различных предметных областях.
Р3	Демонстрировать понимание сущности и значения информации в развитии современного общества, владение основными методами, способами и средствами получения, хранения, переработки информации; использование для решения коммуникативных задач современных технических средств и информационных технологий.
Р4	Выполнять инновационные проекты с применением глубоких профессиональных знаний и эффективных методов проектирования для достижения новых результатов, обеспечивающих конкурентные преимущества в условиях экономических, экологических, социальных и других ограничений.
Р5	Демонстрировать знание о формах организации образовательной и научной деятельности в высших учебных заведениях, иметь навыки преподавательской работы.
Р6	Способность осуществлять организационно-управленческую и

	социально-ориентированную деятельность с соблюдением профессиональной этики
Универсальные компетенции	
Р7	Активно владеть иностранным языком на уровне, позволяющем работать в интернациональной среде, включая разработку документации и представление результатов инновационной деятельности. Толерантность в восприятии социальных и культурных решений.
Р8	Эффективно работать индивидуально, в качестве члена и руководителя группы, состоящей из специалистов различных направлений и квалификаций, демонстрировать ответственность за результаты работы и готовность следовать корпоративной культуре организации.
Р9	Самостоятельно учиться и непрерывно повышать квалификацию в течение всего периода профессиональной деятельности. Способность к интеллектуальному, культурному, нравственному и профессиональному саморазвитию.

РЕФЕРАТ

Выпускная квалификационная работа выполнена на листах машинного текста, содержит 22 рисунка, 15 таблиц, 16 источников, 1 приложение.

Ключевые слова: большие данные, машинное обучение, анализ данных, подготовка данных, дерево решений.

Объектом исследования выпускной квалификационной работы является банковские данные из библиотеки SAS.

Целью исследования выпускной квалификационной работы является разработка алгоритма для предварительной обработки данных с последующим анализом и построением прогнозной модели добросовестных заемщиков.

Область применения: финансовые, аналитические сферы, в дата инженерии.

В результате исследования был разработан алгоритм для анализа и обработки больших данных. По итогу работы разработанный нами алгоритм по анализу и предобработки данных выдал меньшую погрешность нежели с необработанными данными.

Оглавление

Введение

В настоящее время интернет является не только средством связи и получения различных сведений, но и обширным ресурсом, который предоставляет возможность узнать о человеке заочно [1]. Данных собрано такое огромное количество, что процесс предобработки данных многие программисты попросту игнорируют.

BigData - технологии, которые извлекают максимальную пользу из больших данных, однако, исходные данные часто представляются в неподходящем для дальнейшего анализа виде, что может привести к низким показателям точности при дальнейшей работе с ними. В связи с этим предобработка исходных данных является немаловажным этапом работы с данными.

Большие данные (Big Data) условно можно разделить на два уровня: технологический (организация и вывод данных - Data Manager и т.п.) и аналитический (Data Scientist, Data Mining, работа с полученными данными и т.п.) Данная работа проходит на аналитическом уровне и в нем многие эксперты не оговаривают, что большие данные начинаются с какого-то "N" (количество строк или столбцов).

Цель данной работы является разработка алгоритма для предварительной обработки данных с последующим анализом и построением прогнозной модели добросовестных заемщиков.

Достижение такой цели обеспечено **выполнением следующих задач**:

- Разработать алгоритм для предобработки данных;
- Реализовать методику предобработки данных с помощью Python;
- Произвести сравнение над обработанными данными и необработанными;
- Произвести сравнительный анализ результатов работы различных реализаций.

Разрабатываемая информационная система предназначена для

использования в финансовых сферах. Пользователями программы выступают аналитики данных, дата инженеры, отвечающие за процессы сбора, подготовки и очистки данных.

В качестве целевой функции выступает бинарная переменная (GB, good/bad) хороший заемщик (выполняющий свои кредитные обязательства) и плохой заемщик (не выполняющий свои кредитные обязательства), а определяющие (входные) - это анкетные данные, которые и необходимо "чистить".

Таким образом, речь идет о задаче классификации: требуется определить, какому классу: положительному (кредит будет возвращен) или отрицательному (кредит не будет возвращен) – принадлежит клиент.

1. Аналитический обзор разработки алгоритма для анализа и обработки больших данных

1.1 Выбор среды моделирования

Google Colaboratory — это не так давно появившийся облачный сервис, направленный на упрощение исследований в области машинного и глубокого обучения. Colaboratory — это бесплатная среда Jupyter notebook, не требующая настройки и полностью работающая в облаке. Используя Colaboratory, можно писать и выполнять код, сохранять и делиться результатами исследований, а также получать доступ к мощным вычислительным ресурсам (видеокарты и тензорный процессоры). Все эти возможности предоставляются полностью бесплатно [2].

1.1.1 Выбор инструментов моделирования

Для данной работы был выбран язык программирования Python.

В качестве аналитических (программных) конкурентов могут выступить: SAS (дорогая лицензия), SPSS (дорогая лицензия), R (бесплатный, но не очень богатая начинка). В отличие от них Python - бесплатный, и очень богат на наличие библиотек предназначенные для machine learning (машинное обучение).

Python это ещё и простой синтаксис, и удобочитаемость, способствующая быстрому тестированию сложных алгоритмов машинного обучения, процветающее сообщество, поддерживаемое совместными инструментами, такими как Jupyter Notebooks и Google Colab, и широкий выбор библиотек, предназначенных для машинного обучения. Загрузчик Python имеет все это [3].

В работе использованы такие библиотеки, как Pandas и scikit-learn.

Pandas это высокоуровневая Python библиотека для анализа данных, построенная поверх более низкоуровневой библиотеки NumPy [4]. Библиотека pandas предоставляет широкий спектр функций по обработке табличных данных. Библиотека scikit-learn реализует множество алгоритмов машинного обучения. Кроме того, нам понадобится библиотека matplotlib для научной визуализации.

1.2 Выбор среды моделирования

Данные были предоставлены из библиотеки SAS.

SAS - аббревиатура от Statistical Analysis System, что полностью описывает основное направление деятельности компании [5].

1.3 Анализ данных

Согласно описанию рассматриваемой задачи, данные содержат информацию о клиентах, запрашивающих кредит. Всего информации представлено о 3 тысячах клиентов, число признаков о клиенте – 16. Для сохранения конфиденциальности данные обезличены, все значения категориальных признаков заменены символами, а числовые признаки приведены к другому масштабу. Последний столбец содержит символы + и -, соответствующие тому, вернул клиент кредит или нет.

Строки таблицы пронумерованы. Название признаков не имеют каких-либо осмысленных имен и тоже просто пронумерованы (Рисунок 1.).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	b	30.83	0.000	u	g	w	v	1.25	t	t	1	f	g	202.0	0	+
1	a	58.67	4.460	u	g	q	h	3.04	t	t	6	f	g	43.0	560	+
2	a	24.50	0.500	u	g	q	h	1.50	t	f	0	f	g	280.0	824	+
3	b	27.83	1.540	u	g	w	v	3.75	t	t	5	t	g	100.0	3	+
4	b	20.17	5.625	u	g	w	v	1.71	t	f	0	f	s	120.0	0	+

Рисунок 1 – Верхняя часть данных

Для удобства зададим столбцам имена (Рисунок 2.).

	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13	A14	A15	class
0	b	30.83	0.000	u	g	w	v	1.25	t	t	1	f	g	202.0	0	+
1	a	58.67	4.460	u	g	q	h	3.04	t	t	6	f	g	43.0	560	+
2	a	24.50	0.500	u	g	q	h	1.50	t	f	0	f	g	280.0	824	+
3	b	27.83	1.540	u	g	w	v	3.75	t	t	5	t	g	100.0	3	+
4	b	20.17	5.625	u	g	w	v	1.71	t	f	0	f	s	120.0	0	+

Рисунок 2 – Данные, с заданными именами признаков

С помощью метода describe() получим некоторую сводную информацию по всей таблице. По умолчанию будет выдана информация только для количественных признаков. Это общее их количество (count), среднее значение (mean), стандартное отклонение (std), минимальное (min), максимальное (max) значения, медиана (50%) и значения нижнего (25%) и верхнего (75%) квартилей (Рисунок 3.).

	A2	A3	A8	A11	A14	A15
count	2948.000000	3000.000000	3000.000000	3000.000000	2945.000000	3000.000000
mean	31.787259	4.851932	2.340435	2.555000	182.701868	1056.840000
std	11.983570	5.001877	3.440362	5.064141	173.275141	5227.325782
min	13.750000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	22.670000	1.030000	0.198750	0.000000	70.000000	0.000000
50%	28.580000	3.000000	1.000000	0.000000	160.000000	5.000000
75%	38.580000	7.540000	3.000000	3.000000	274.000000	447.000000
max	80.250000	28.000000	28.500000	67.000000	2000.000000	100000.000000

Рисунок 3 – сводная информация для количественных признаков

Выделим числовые и категориальные признаки (Рисунок 4.).

```
17
18 categorical_columns = [c for c in data.columns if data[c].dtype.name == 'object']
19 numerical_columns   = [c for c in data.columns if data[c].dtype.name != 'object']
20 print (categorical_columns)
21 print (numerical_columns)
```

Рисунок 4 – Выделение числовых и категориальных признаков

Теперь мы можем получить некоторую общую информацию по категориальным признакам (Рисунок 5.).

	A1	A4	A5	A6	A7	A9	A10	A12	A13	class
count	2952	2975	2975	2963	2963	3000	3000	3000	3000	3000
unique	2	3	3	14	9	2	2	2	3	2
top	b	u	g	c	v	t	f	f	g	-
freq	2034	2270	2270	590	1731	1675	1671	1612	2722	1579

Рисунок 5 – Сводная информация по категориальным признакам

В таблице для каждого категориального признака приведено общее число заполненных ячеек (count), количество значений, которые принимает данный признак (unique), самое популярное (часто встречающееся) значение этого признака (top) и количество объектов, в которых встречается самое частое значение данного признака (freq).

Функция scatter_matrix из модуля pandas.tools.plotting позволяет построить для каждой количественной переменной гистограмму, а для каждой пары таких переменных – диаграмму рассеяния (Рисунок 6.).

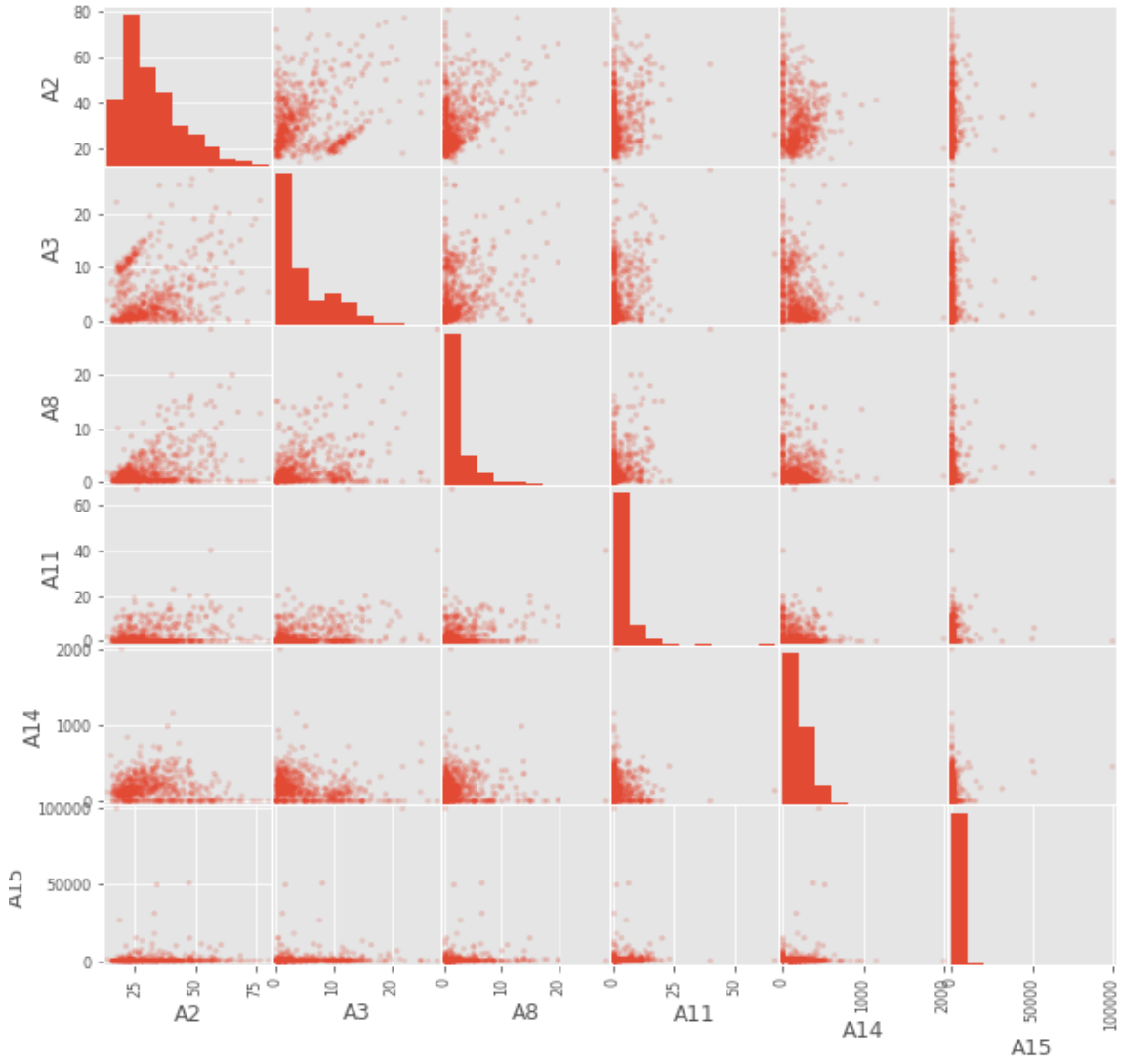

Рисунок 6 – Гистограммы и диаграммы

Из построенных диаграмм видно, что признаки не сильно коррелируют между собой, что, впрочем, можно также легко установить, посмотрев на

корреляционную матрицу (Рисунок 7.). Все ее недиагональные значения по модулю не превосходят значение экспертной оценки в 0.4.

	A2	A3	A8	A11	A14	A15
A2	1.000000	0.209883	0.409926	0.193464	-0.081348	0.027701
A3	0.209883	1.000000	0.301779	0.280811	-0.230784	0.111900
A8	0.409926	0.301779	1.000000	0.325692	-0.082149	0.056180
A11	0.193464	0.280811	0.325692	1.000000	-0.123779	0.063181
A14	-0.081348	-0.230784	-0.082149	-0.123779	1.000000	0.068076
A15	0.027701	0.111900	0.056180	0.063181	0.068076	1.000000

Рисунок 7 – Корреляционная матрица

2. Предобработка и подготовка данных

2.1 Подготовка данных

Узнаём количество пропущенных значений с помощью метода **count** (Рисунок 8.).

```
A1       2952
A2       2948
A3       3000
A4       2975
A5       2975
A6       2963
A7       2963
A8       3000
A9       3000
A10      3000
A11      3000
A12      3000
A13      3000
A14      2945
A15      3000
class    3000
```

Рисунок 8 – Общее количество заполненных ячеек

Если данные содержат пропущенные значения, то имеются два способа решения этой проблемы:

- удалить столбцы с такими значениями
- удалить строки с такими значениями

После этого, к сожалению, данных может стать совсем мало, поэтому рассмотрим простые альтернативные способы.

Количественные признаки

Заполнить пропущенные значения можно с помощью метода библиотеки Pandas fillna (заполняет значения NA/NaN используя заданные методы). Заполним медианными значениями.

Категориальные признаки

Теперь рассмотрим пропущенные значения в столбцах, соответствующих категориальным признакам. Простая стратегия – заполнение пропущенных значений самым популярным в столбце.

К примеру, возьмём первый категориальный столбец **A1**:

```
           A1
count    2952
unique      2
top         b
freq     2034
```

В столбце **A1** имеются пропущенные значения. Наиболее частым (встречается 2034 раз) является b. Заполняем все пропуски этим значением

```
data['A1'] = data['A1'].fillna('b')
```

Автоматизируем процесс:

```
data_describe = data.describe(include=[object])
for c in categorical_columns:
    data[c] = data[c].fillna(data_describe[c]['top'])
```

Теперь все элементы заполнены.

2.1.1 Векторизация

Так как подключенная библиотека «scikit-learn» не умеет напрямую обрабатывать категориальные признаки. Поэтому прежде чем подавать данные на вход алгоритмов машинного обучения преобразуем категориальные признаки в количественные.

Категориальные признаки, принимающие два значения (т.е. бинарные признаки) и принимающие большее количество значений будем обрабатывать по-разному.

Вначале выделим бинарные и не бинарные признаки:

```
binary_columns    = [c for c in categorical_columns if
data_describe[c]['unique'] == 2]
nonbinary_columns = [c for c in categorical_columns if
data_describe[c]['unique'] > 2]
print binary_columns, nonbinary_columns
```

```
['A1', 'A9', 'A10', 'A12', 'class' ] - бинарные признаки
['A4', 'A5', 'A6', 'A7', 'A13'] - не бинарные признаки
```

Бинарные признаки

Значения бинарных признаков просто заменим на 0 и 1

```
for c in binary_columns[1:]:
    top = data_describe[c]['top']
    top_items = data[c] == top
```

```
    data.loc[top_items, c] = 0
    data.loc[np.logical_not(top_items), c] = 1
```

Не бинарные признаки

К не бинарными признакам применим метод векторизации, который заключается в следующем.

Признак j, принимающий s значений, заменим на s признаков, принимающих значения 0 или 1, в зависимости от того, чему равно значение исходного признака j.

Например, в нашей задаче признак A4 принимает 3 различных значения:

```
data['A4'].unique()
array(['y' 'u' 'l'], dtype=object)
```

Заменим признак A4 тремя признаками: A4_u, A4_y, A4_l.

- Если признак A4 принимает значение u, то признак A4_u равен 1, A4_y равен 0, A4_l равен 0.
- Если признак A4 принимает значение y, то признак A4_y равен 0, A4_y равен 1, A4_l равен 0.
- Если признак A4 принимает значение l, то признак A4_l равен 0, A4_y равен 0, A4_l равен 1.

2.1.2 Нормализация количественных признаков

Многие алгоритмы машинного обучения чувствительны к масштабированию данных. В этом случае количественные признаки полезно *нормализовать*. Это можно делать разными способами. Например, каждый количественный признак приведем к нулевому среднему и единичному среднеквадратичному отклонению (Рисунок 9.).

```
data_numerical = data[numerical_columns]
data_numerical = (data_numerical - data_numerical.mean()) /
data_numerical.std()
data_numerical.describe()
```

	A2	A3	A8	A11	A14
count	3.000000e+03	3.000000e+03	3.000000e+03	3.000000e+03	3.000000e+03
mean	3.855434e-16	-2.890133e-15	5.324051e-16	4.168517e-16	2.227848e-17
std	1.000000e+00	1.000000e+00	1.000000e+00	1.000000e+00	1.000000e+00
min	-1.512768e+00	-9.700222e-01	-6.802875e-01	-5.045278e-01	-1.061615e+00
25%	-7.623431e-01	-7.640995e-01	-6.225174e-01	-5.045278e-01	-6.364700e-01
50%	-2.651445e-01	-3.702473e-01	-3.896204e-01	-5.045278e-01	-1.297897e-01
75%	5.483767e-01	5.374119e-01	1.917139e-01	8.787275e-02	5.224883e-01
max	4.081768e+00	4.627876e+00	7.603725e+00	1.272575e+01	1.058621e+01

	A15
count	3.000000e+03
mean	1.902552e-16
std	1.000000e+00
min	-2.021760e-01
25%	-2.021760e-01
50%	-2.012195e-01
75%	-1.166639e-01
max	1.892806e+01

Рисунок 9 – Информация о масштабирование количественных признаков

Соединив всё в одну таблицу можем приступать к машинному обучению

3. Моделирование алгоритма с помощью машинного обучения

3.1 Машинное обучение

Обучающая и тестовая выборки

Обучаться модель будет на обучающей выборке, а проверка качества осуществлять с помощью тестовой выборки.

В данной задаче разобьем имеющиеся данные на обучающую и тестовую выборки в соотношение (70% / 30%).

```
from sklearn.cross_validation import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size = 0.3, random_state = 11)

N_train, _ = X_train.shape
N_test,  _ = X_test.shape
```

X_train, y_train – это обучающая выборка, X_test, y_test – тестовая.

Алгоритм машинного обучения

В библиотеке «scikit-learn» реализована масса алгоритмов машинного обучения. Самым удобным для себя я считаю алгоритм «Random Forest» и буду использовать именно его.

Это один из самых популярных методов. Его реализация заключается в построение ансамбля случайных деревьев, каждое из которых обучается на выборке, полученной из исходной с помощью процедуры изъятия с возвращением (Рисунок 10.).

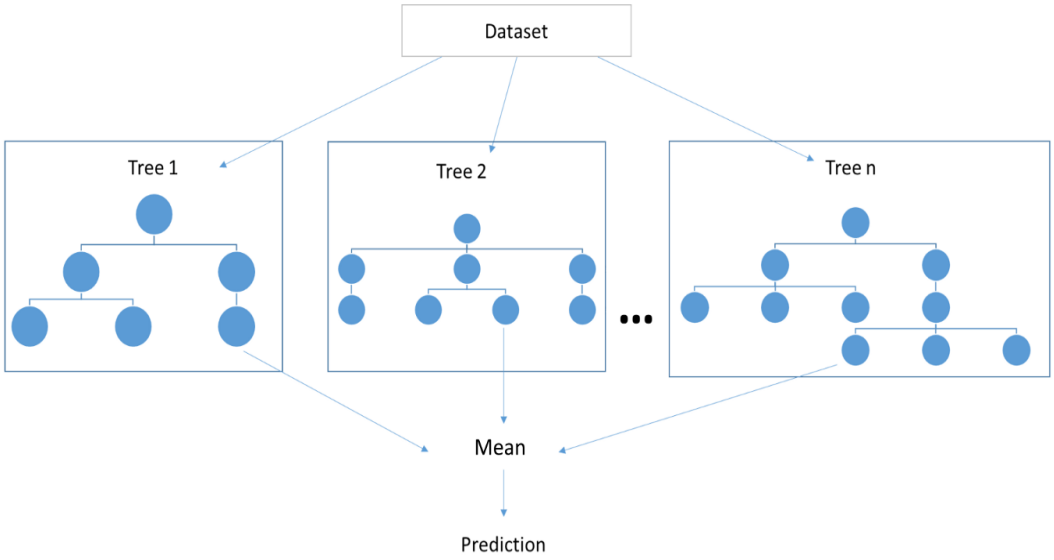

Рисунок 10 – Процесс работы метода RandomForest

3.1.1 Описание концептуальной модели метода RandomForest

Построение концептуальной модели предшествует этапу программирования имитационной модели.

Перейдём к описанию модели:

1. **Модель компьютерная.** В роли модели выступает программный код.

2. **Модель линейная.** Как следует из определения, решающее дерево a(x) разбивает всё признаковое пространство на некоторое количество непересекающихся подмножеств {J1, . . ., Jn}, и в каждом подмножестве Jj выдаёт константный прогноз Wj. Значит соответствующий алгоритм можно записать аналитически a(x)=Xn.

3. **Модель статичная.** Не изменяется во времени

4. **Модель детерминированная.** Случайные эффекты не задействованы, прогнозируется всё на реальных данных, аналитически-математическими методами.

5. **Модель имитационная.** исследуются математические модели в виде алгоритмов, воспроизводящего функционирование исследуемой системы

путём последовательного выполнения большого количества элементарных операций.

3.1.2 Описание модели RandomForestClassifier в colab

```
RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
            max_depth=None, max_features='auto', max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=1, min_samples_split=2,
            min_weight_fraction_leaf=0.0, n_estimators=100, n_jobs=None,
            oob_score=False, random_state=11, verbose=0, warm_start=False)
```

Bootstrap: boolean, optional (default=True)

Порог для ранней остановки роста деревьев. Узел расколется, если его примесь выше порога, в противном случае это лист.

class_weight: (default=None)

Веса, связанные с классами в форме. Если(None), все классы должны иметь один вес.

criterion: string, optional (default=" gini")

Функция для измерения качества раскола.

max_depth: integer or None, optional (default=None)

Максимальная глубина дерева. Если None, то узлы расширяются до тех пор, пока все листья не станут чистыми или пока все листья не будут содержать меньше чем min_samples_split samples.

max_features: int, float, string or None, optional (default=" auto")

Количество функций, которые следует учитывать при поиске наилучшего разделения

Если «auto», то max_features = sqrt (n_features).

max_leaf_nodes: int or None, optional (default=None)

Лучшие узлы определяются как относительное уменьшение примесей. Если None, то неограниченное количество листовых узлов.

min_impurity_decrease: float, optional (default=0.)

Узел будет разделен, если это разделение вызовет уменьшение примеси, большее или равное этому значению.

min_impurity_split: float, (default<auto>=1e-7)

Порог для ранней остановки роста деревьев. Узел разделится, если его примесь выше порога, в противном случае это лист.

min_samples_leaf: int, float, optional (default=1)

Какое минимальное количество образцов должно быть в листовом узле.

min_samples_split: int, float, optional (default=2)

Минимальное количество выборок, необходимое для разделения внутреннего узла.

min_weight_fraction_leaf: float, optional (default=0.)

Минимальная взвешенная доля общей суммы весов (всех входных выборок), необходимая для конечного узла.

n_estimators: integer, optional (default=10)

Количество деревьев в лесу.

n_jobs: int or None, optional (default=None)

Количество заданий, выполняемых параллельно, как для подгонки, так и для прогнозирования

oob_score: bool (default=False)

Использовать ли образцы из пакета для оценки точности обобщения.

random_state: int, RandomState instance or None, optional (default=None)

Если int, random_state - начальное число, используемое генератором случайных чисел.

verbose: int, optional (default=0)

Управляет многословием при подборе и прогнозировании.

warm_start: bool, optional (default=False)

Если задано значение True, повторно использовать решение предыдущего вызова для подгонки и добавить дополнительные оценки в ансамбль, в противном случае просто подгонять целый новый лес.

Приступим к обучению:

```
from sklearn import ensemble
rf = ensemble.RandomForestClassifier(n_estimators=100,
random_state=11)
rf.fit(X_train, y_train)
```

После того, как модель обучена, мы можем предсказывать значение целевого признака по входным признакам для новых объектов. Делается это с помощью метода predict.

Нас интересует качество построенной модели, поэтому будем предсказывать значение выходного признака на тех данных, для которых оно известно: на обучающей и (что более важно) тестовой выборках:

```
err_train = np.mean(y_train != rf.predict(X_train))
err_test  = np.mean(y_test  != rf.predict(X_test))
print err_train, err_test
```

```
0.0 – обучающаяся (0.0%)
0.0044444444444444444 – тестовая (0.4%)
```

Итак, погрешность на тестовой выборке составила **0.4%**, а на обучающейся и вовсе **0%**. Посмотрим на это наглядно.

3.1.3 Отбор признаков с помощью алгоритма случайного леса

Одна из важных процедур предобработки данных в алгоритмах их анализа является отбор значимых признаков. Его цель заключается в том, чтобы отобрать наиболее существенные признаки для решения рассматриваемой задачи классификации.

Отбор признаков необходим для следующих целей:

- Для лучшего понимания задачи. Человеку легче разобраться с небольшим количеством признаков, чем с огромным их количеством.
- Для ускорения алгоритмов.
- Для улучшения качества предсказания. Устранение шумовых признаков может уменьшить ошибку алгоритма на тестовой выборке, т.е. улучшить качество предсказания.

```
importances = rf.feature_importances_
indices = np.argsort(importances)[::-1]

print("Feature importances:")
for f, idx in enumerate(indices):
```

```
    print("{:2d}. feature '{:5s}' ({:.4f})".format(f + 1,
feature_names[idx], importances[idx]))
```

```
Feature importances:
 1. feature 'A9   ' (0.2610)
 2. feature 'A8   ' (0.0994)
 3. feature 'A11  ' (0.0932)
 4. feature 'A3   ' (0.0794)
 5. feature 'A15  ' (0.0782)
 6. feature 'A14  ' (0.0706)
 7. feature 'A2   ' (0.0662)
 8. feature 'A10  ' (0.0504)
 9. feature 'A1   ' (0.0144)
10. feature 'A7_h ' (0.0133)
11. feature 'A12  ' (0.0130)
12. feature 'A6_x ' (0.0111)
13. feature 'A6_c ' (0.0099)
14. feature 'A7_ff' (0.0092)
15. feature 'A6_ff' (0.0090)
16. feature 'A7_v ' (0.0089)
17. feature 'A6_k ' (0.0087)
18. feature 'A6_q ' (0.0079)
19. feature 'A5_g ' (0.0078)
20. feature 'A6_w ' (0.0078)
21. feature 'A4_y ' (0.0076)
22. feature 'A4_u ' (0.0075)
23. feature 'A5_p ' (0.0075)
24. feature 'A6_aa' (0.0069)
25. feature 'A13_g' (0.0069)
26. feature 'A6_cc' (0.0065)
27. feature 'A6_i ' (0.0065)
28. feature 'A7_bb' (0.0050)
29. feature 'A13_s' (0.0049)
30. feature 'A13_p' (0.0045)
31. feature 'A6_d ' (0.0040)
32. feature 'A6_m ' (0.0036)
33. feature 'A7_j ' (0.0022)
34. feature 'A6_e ' (0.0019)
35. feature 'A5_gg' (0.0012)
36. feature 'A6_j ' (0.0009)
37. feature 'A7_z ' (0.0009)
38. feature 'A4_l ' (0.0008)
39. feature 'A7_dd' (0.0008)
40. feature 'A7_o ' (0.0004)
41. feature 'A6_r ' (0.0002)
42. feature 'A7_n ' (0.0001)
```

Рисунок 11 – Список признаков

Построим столбцевую диаграмму, графически представляющую значимость первых 20 признаков и сразу же выведем первые 8 признаков, оказывающих наибольшее влияние (Рисунок 11.).

```
d_first = 20
plt.figure(figsize=(8, 8))
plt.title("Feature importances")
plt.bar(range(d_first), importances[indices[:d_first]],
align='center')
plt.xticks(range(d_first),
np.array(feature_names)[indices[:d_first]], rotation=90)
plt.xlim([-1, d_first]);
```

```
best_features = indices[:8]
best_features_names = feature_names[best_features]
print(best_features_names)
```

```
Index(['A9', 'A8', 'A11', 'A3', 'A15', 'A14', 'A2', 'A10'], dtype='object')
```

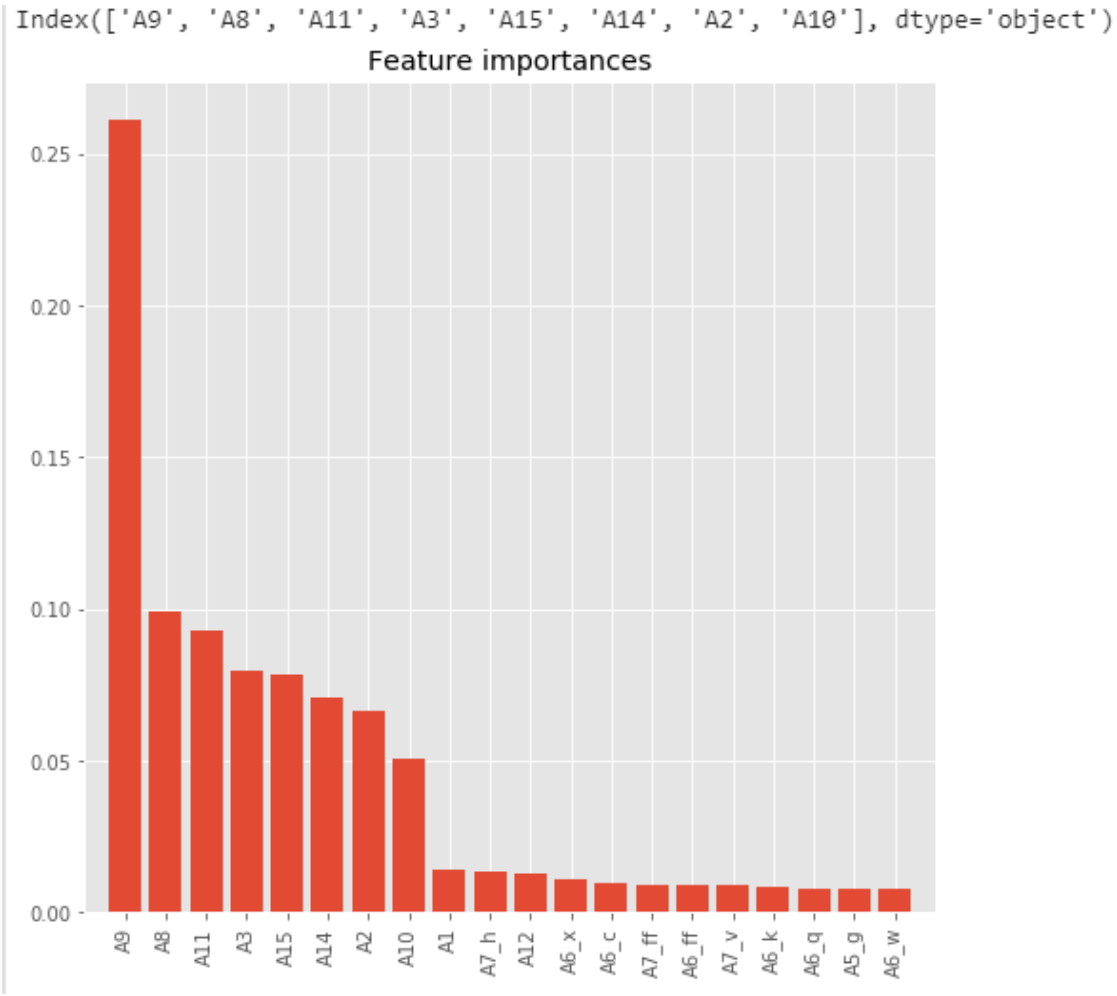

Рисунок 12 – Диаграмма значимых признаков

На диаграмме (Рисунок 12.) хорошо видно, какие именно признаки оказывают наибольшее влияние. Теперь будем использовать только эти признаки для обучения модели.

3.1.4 Проверка результатов обучения

Проверяем модель встроенными функциями библиотек «Python»:

```
from sklearn.metrics import confusion_matrix
import itertools
```

Запускаем проверку на обучающей выборке:

```
cm = confusion_matrix(lab_enc.fit_transform(y_train[:3000]),
rf.predict(X_train[:3000]))
```

```
plot_confusion_matrix(cm, classes = ['positive', 'negative'],
                      title = 'God-Bad Matrix')
```

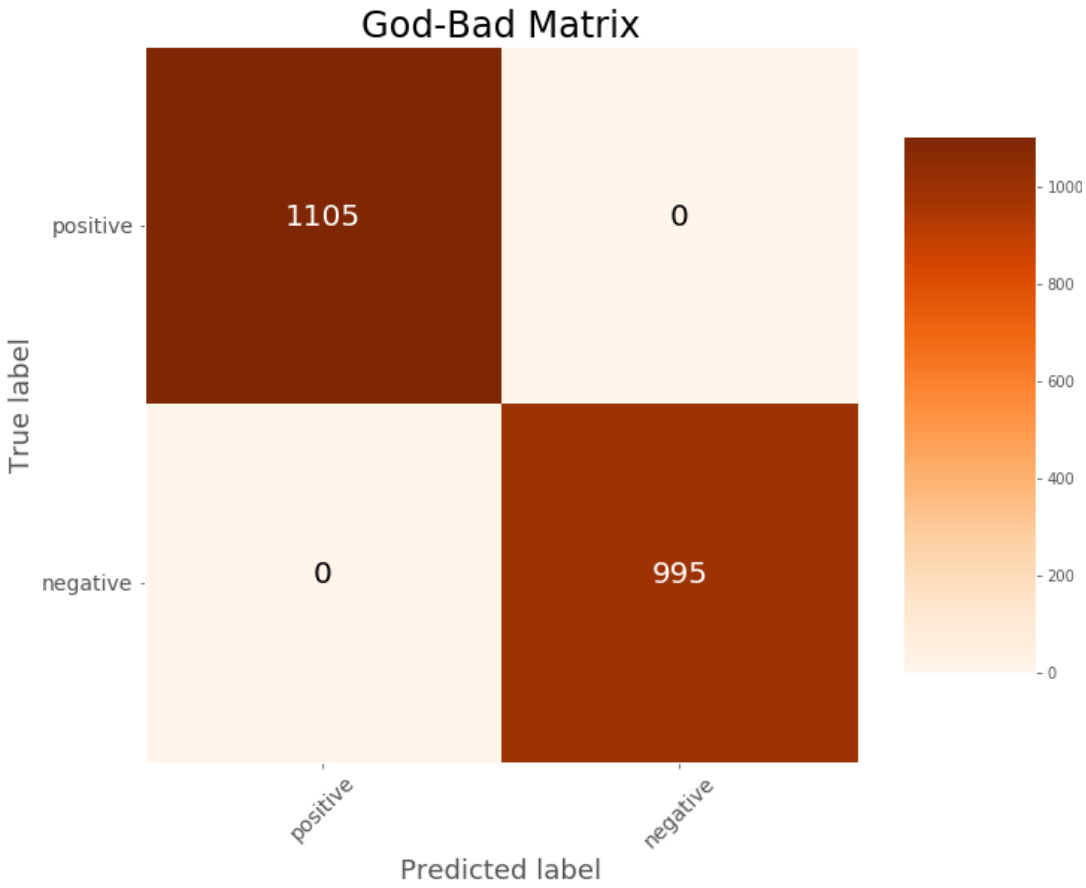

Рисунок 13 – Матрица результатов (обучающиеся данные)

На обучающей выборке модель выдала отличный результат с погрешностью в 0%, как и было получено до этого.

На матрице (Рисунок 13.) наглядно видно, что из 1105 правильно выданных кредитов, модель выдала все 1105 положительных результатов и из 995 не выданных кредитов, модель также выдала 995 отрицательных результатов.

Видно, что модель хорошо обучилась. Проверим правильность результатов на тестовой выборке.

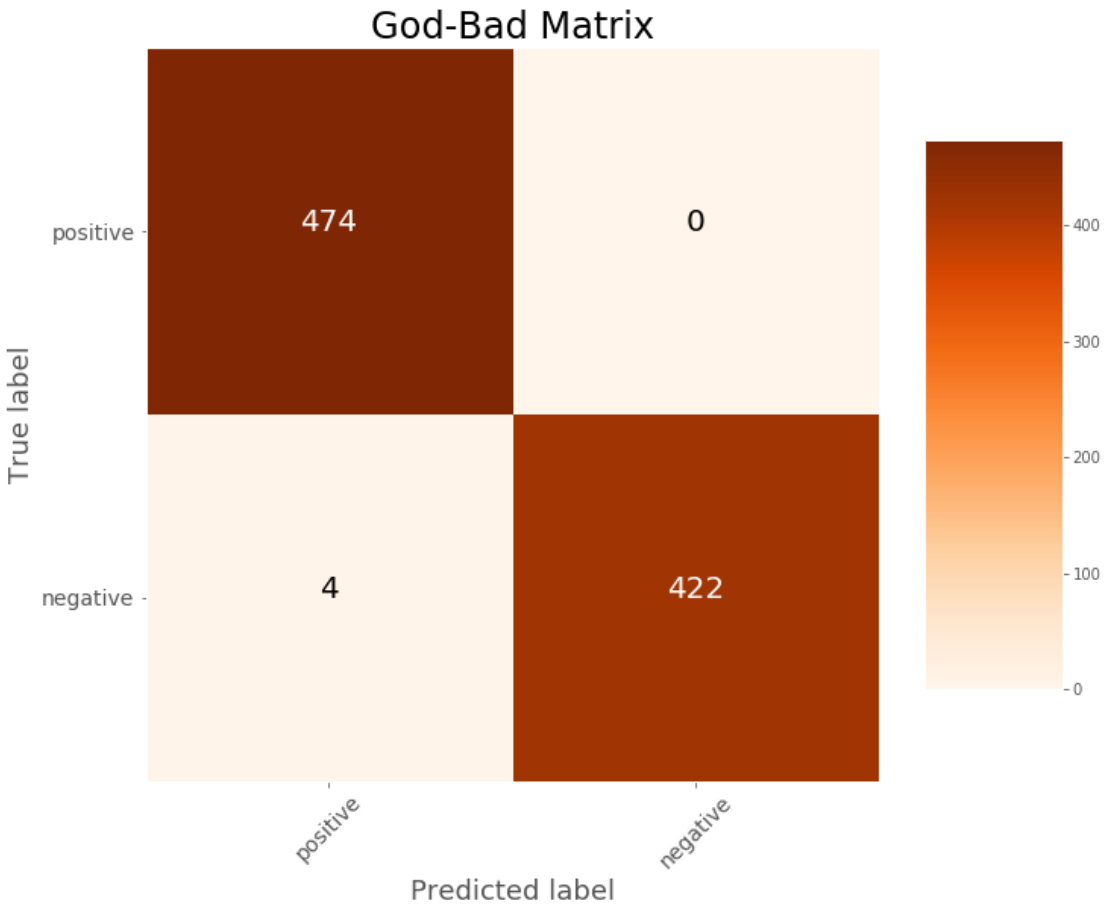

Рисунок 14 – Матрица результатов (тестовые данные)

На тестовой выборке (Рисунок 14) модель показала себя чуть хуже, но всё же результат отличный, из 900 тестов всего 4 оказались неверными, а это означает, что погрешность составила всего 0,4%.

3.1.5 Результат работы с необработанными данными

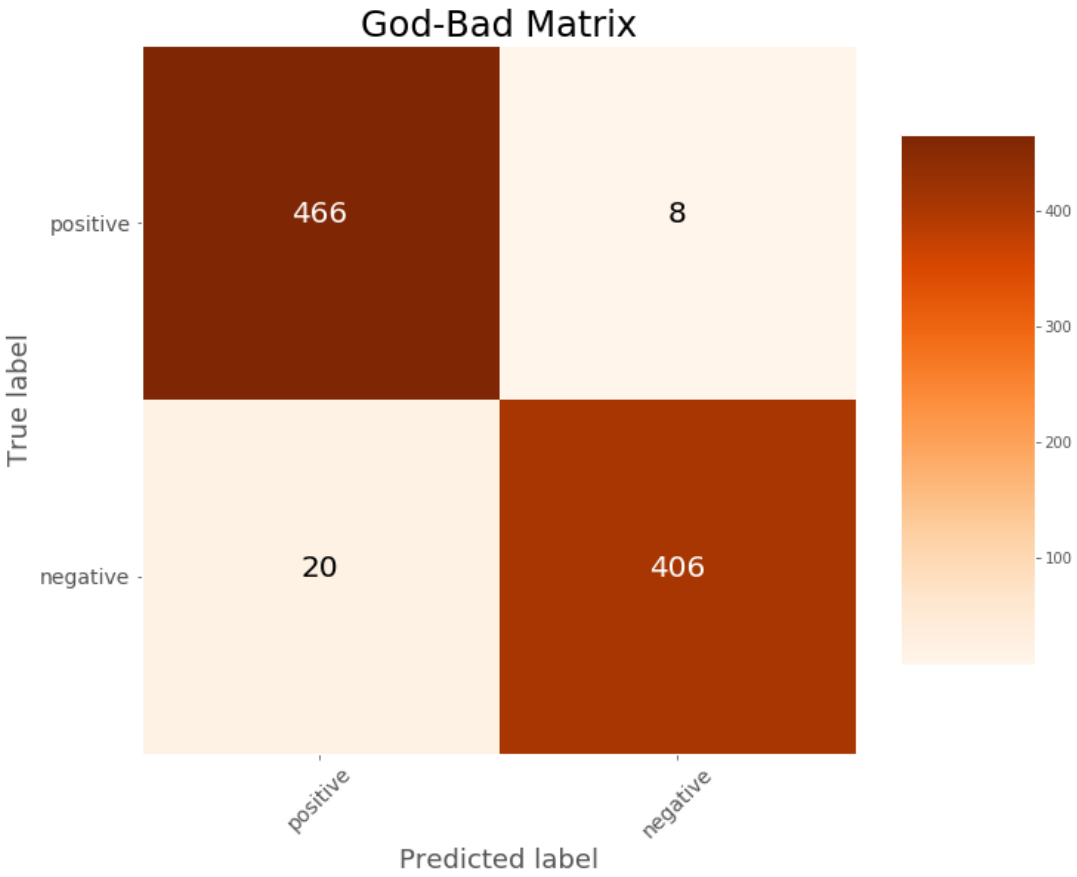

Рисунок 15 – Матрица результатов (тестовые данные)

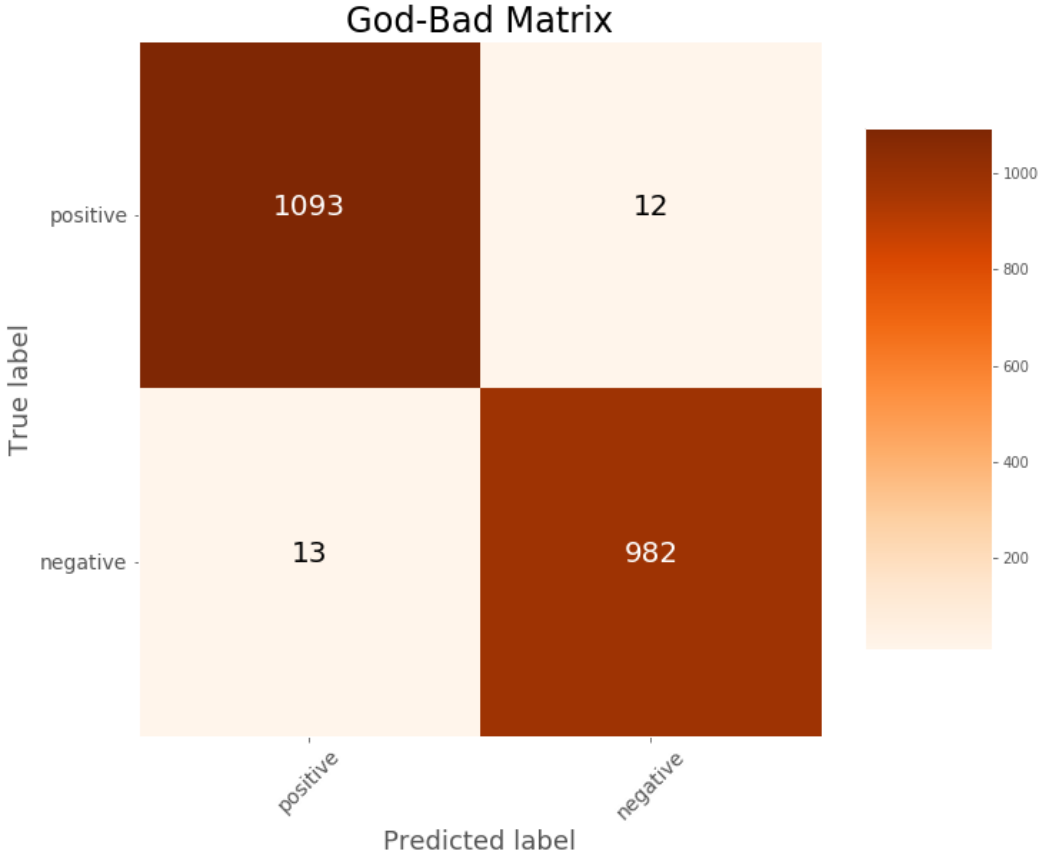

Рисунок 16 – Матрица результатов (обучающиеся данные)

```
0.011904761904761904 – обучающаяся (1.2%)
0.03111111111111111 – тестовая (3.1%)
```

3.1.6 Пояснение работы ML «RandomForest»

Для наглядности работы модели RandomForestClassifier, можем визуализировать дерево решений, чтобы наше решение не было каким-то необъяснимым «чёрным ящиком». Если посмотреть на отдельное дерево решений (Рисунок 17 - 18), мы увидим, что эта модель – это не необъяснимый метод, а последовательность логических вопросов и ответов – так же, как мы формировали бы при прогнозировании.

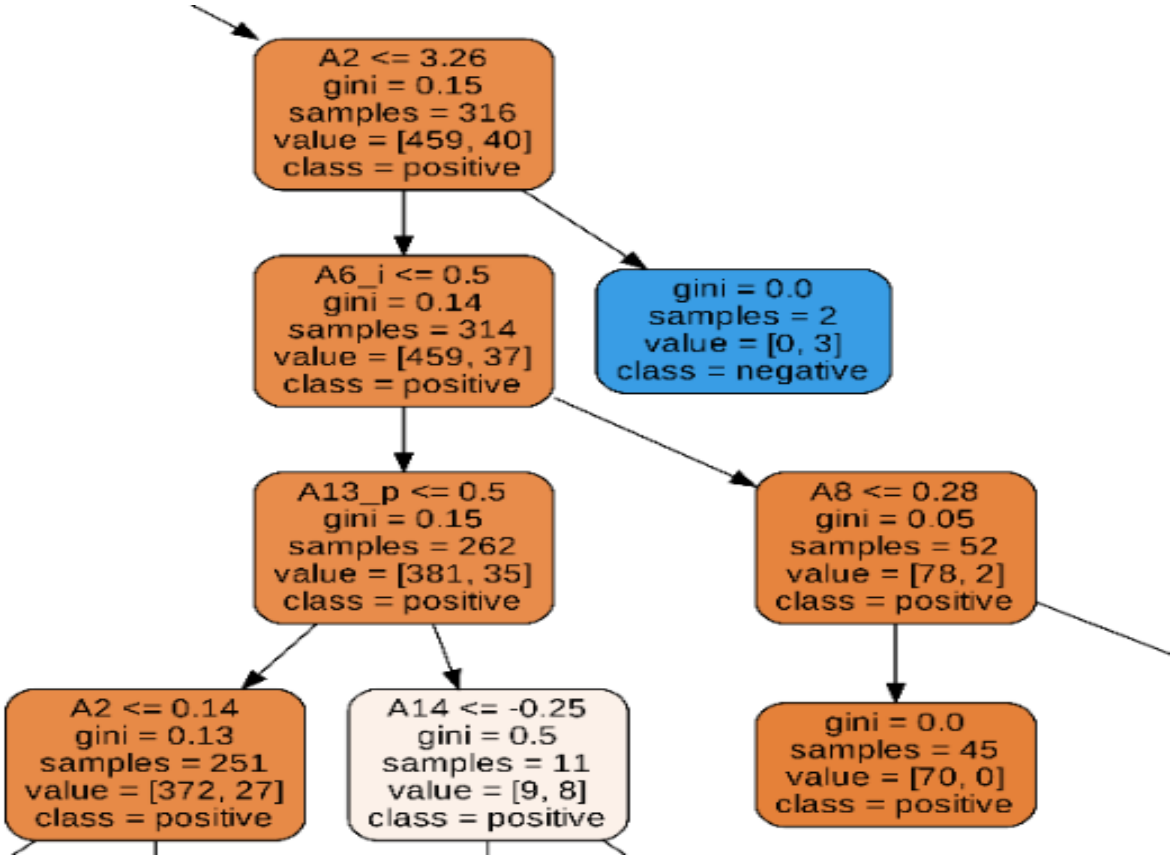

Рисунок 17 – Параметры отдельных листьев

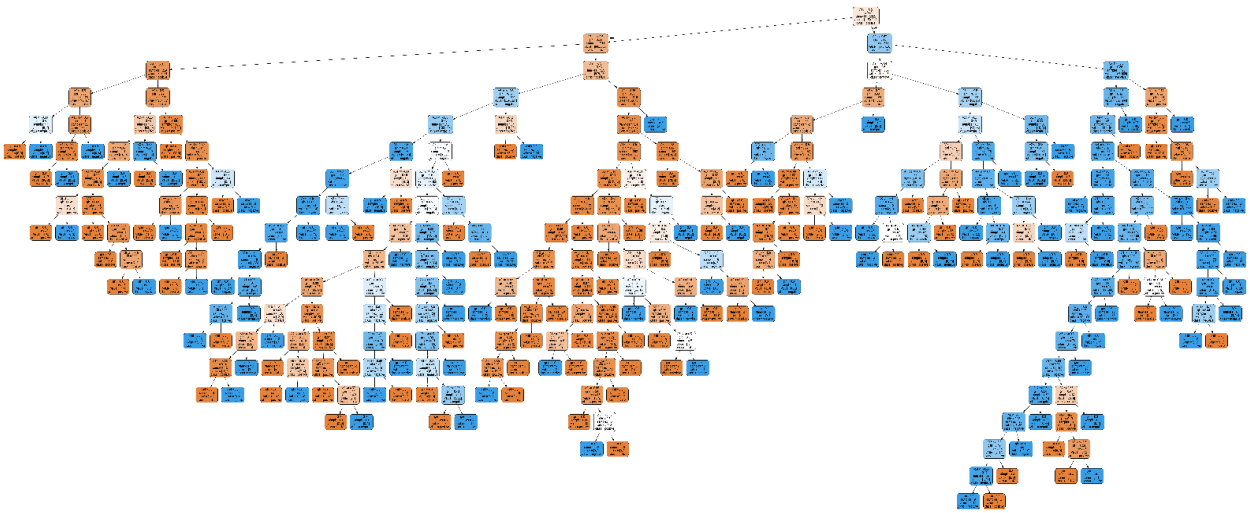

Рисунок 18 – Полное дерево

4. Сравнение реализаций

Механизм важности с опущенным столбцом

Идея состоит в том, чтобы получить базовый показатель производительности, как при важности перестановок, но затем полностью удалить столбец, переобучить модель и пересчитать показатель производительности. Значение важности функции - это разница между базовой линией и оценкой модели, в которой отсутствует эта функция. Эта стратегия отвечает на вопрос о том, насколько важна функция для общей производительности модели

Значения важности могут отличаться между двумя атрибутами, но порядок значений признаков должен быть примерно одинаковым.

Важность перестановки не требует переобучения базовой модели для измерения влияния перемешивания переменных на общую точность модели. Поскольку обучение модели может быть чрезвычайно долгим и даже занимать дни, это большой выигрыш в производительности. Риск - это потенциальный уклон в сторону коррелированных прогностических переменных.

```
Feature imp:
 1. feature 'A9   ' (0.2562)
 2. feature 'A8   ' (0.0233)
 3. feature 'A15  ' (0.0224)
 4. feature 'A3   ' (0.0186)
 5. feature 'A14  ' (0.0152)
 6. feature 'A11  ' (0.0105)
 7. feature 'A2   ' (0.0105)
 8. feature 'A10  ' (0.0057)
 9. feature 'A6_c ' (0.0038)
10. feature 'A13_p' (0.0019)
11. feature 'A6_x ' (0.0019)
12. feature 'A6_k ' (0.0014)
13. feature 'A1   ' (0.0014)
14. feature 'A6_w ' (0.0014)
15. feature 'A7_h ' (0.0014)
16. feature 'A13_g' (0.0010)
17. feature 'A4_u ' (0.0010)
18. feature 'A6_m ' (0.0010)
19. feature 'A4_y ' (0.0005)
20. feature 'A12  ' (0.0005)
21. feature 'A7_v ' (0.0005)
22. feature 'A6_ff' (0.0005)
23. feature 'A6_i ' (0.0005)
24. feature 'A6_q ' (0.0005)
25. feature 'A5_g ' (0.0005)
26. feature 'A7_ff' (0.0005)
27. feature 'A5_gg' (0.0005)
28. feature 'A4_l ' (0.0000)
29. feature 'A13_s' (0.0000)
30. feature 'A5_p ' (0.0000)
31. feature 'A6_aa' (0.0000)
32. feature 'A6_cc' (0.0000)
33. feature 'A6_d ' (0.0000)
34. feature 'A6_j ' (0.0000)
35. feature 'A6_r ' (0.0000)
36. feature 'A7_bb' (0.0000)
37. feature 'A7_dd' (0.0000)
38. feature 'A7_j ' (0.0000)
39. feature 'A7_n ' (0.0000)
40. feature 'A7_o ' (0.0000)
41. feature 'A7_z ' (0.0000)
42. feature 'A6_e ' (0.0000)
Index(['A9', 'A8', 'A15', 'A3', 'A14', 'A11', 'A2', 'A10'], dtype='object')
```

Рисунок 19 – Важные параметры механизма с опущенным столбцом

На (Рисунок 19) видно, что признак A9, оказывает наибольшее влияние на прогнозирование нежели остальные признаки.

Диаграмма позволяет нам в этом наглядно убедиться (Рисунок 20) и стоит заметить, что расположение признаков по значимости отличается от предыдущего метода.

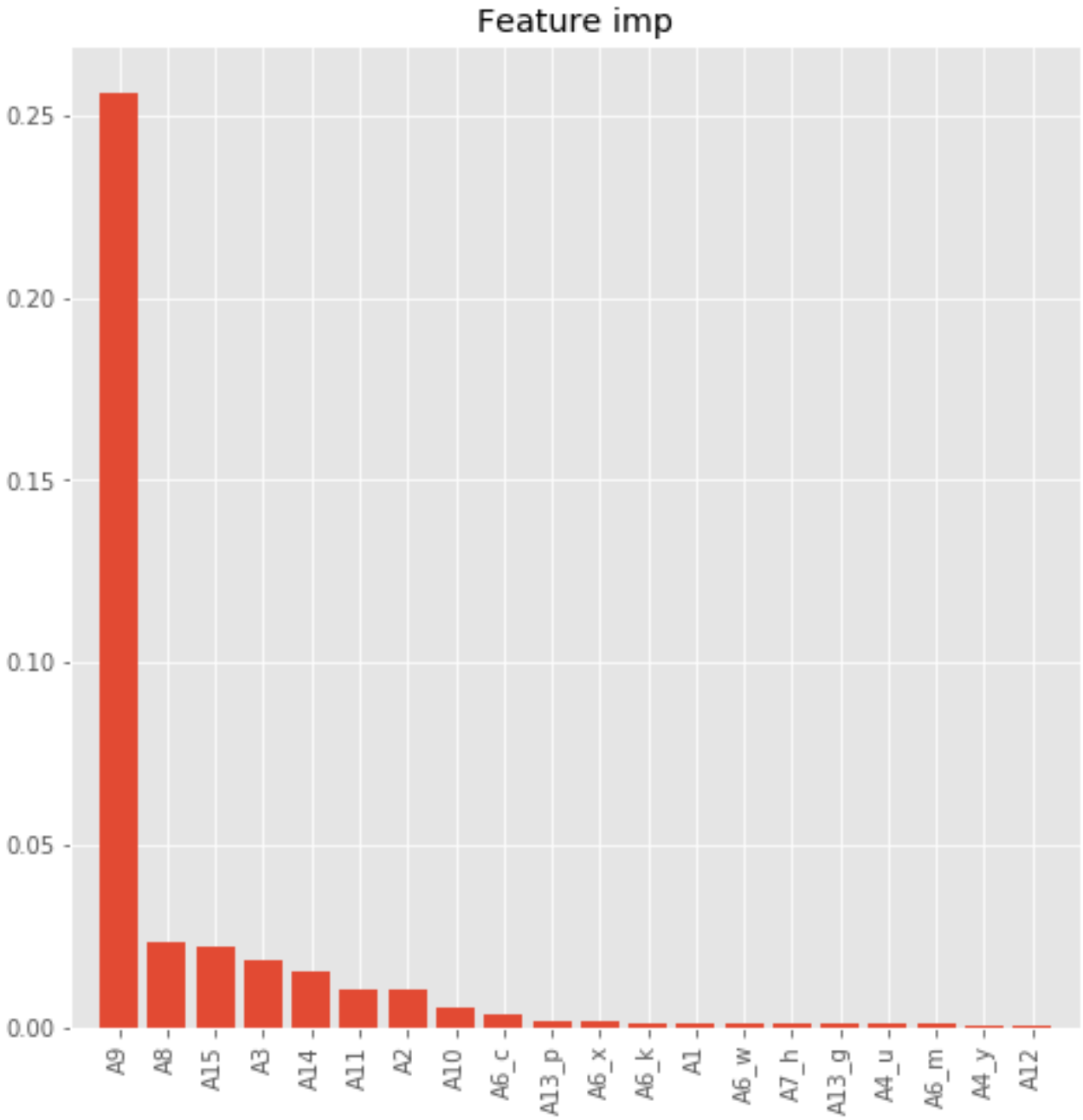

Рисунок 20 – Диаграмма значимых признаков механизма с опущенным столбцом

Обучение происходило на необработанных данных для большей наглядности.

Как мы можем заметить (Таблица 1), погрешность результатов меньше, чем в предыдущем механизме выделение важных атрибутов.

Таблица 1 – сравнения погрешностей на необработанных данных

Механизм	Обучающиеся данные	Тестовые данные
Стандарт. Механизм RF	1,2%	3,1%
Механизм с опущенным столбцом	0,14%	1%

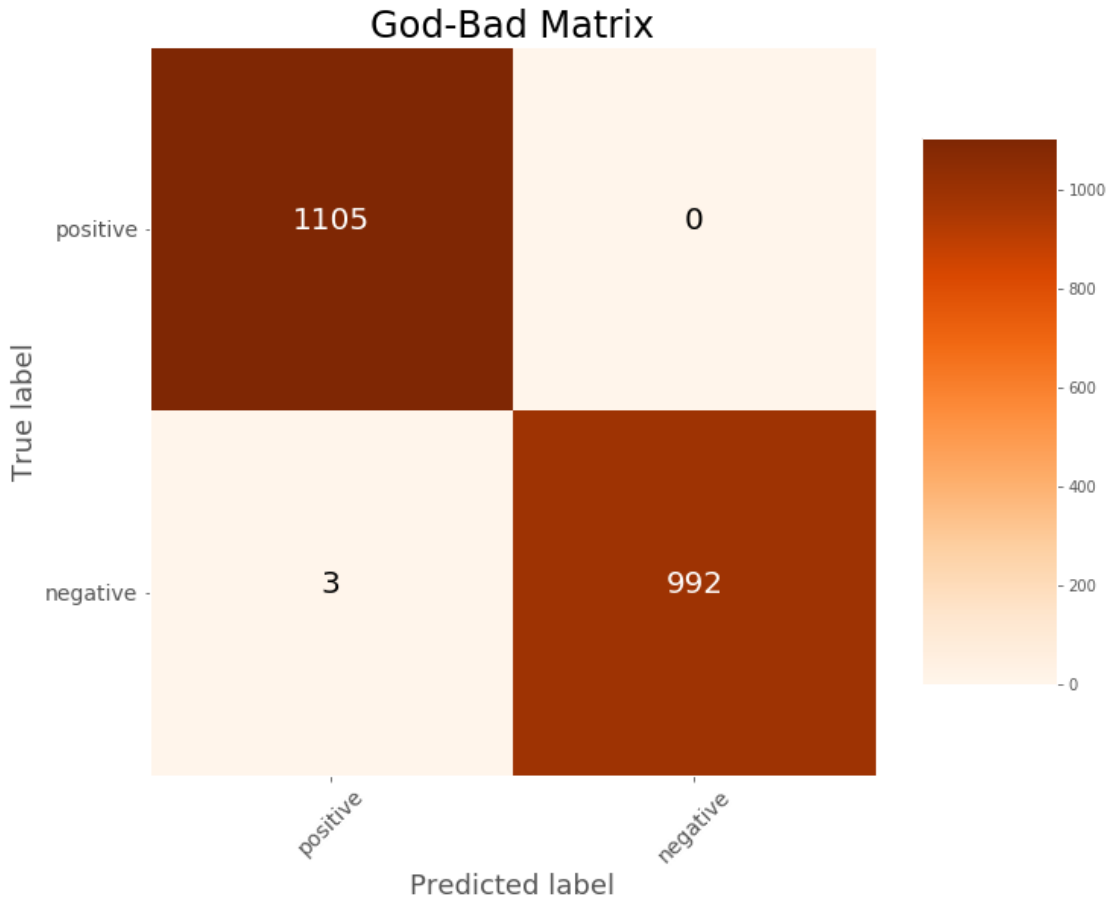

Рисунок 21 – Матрица результатов (обучающиеся данные)

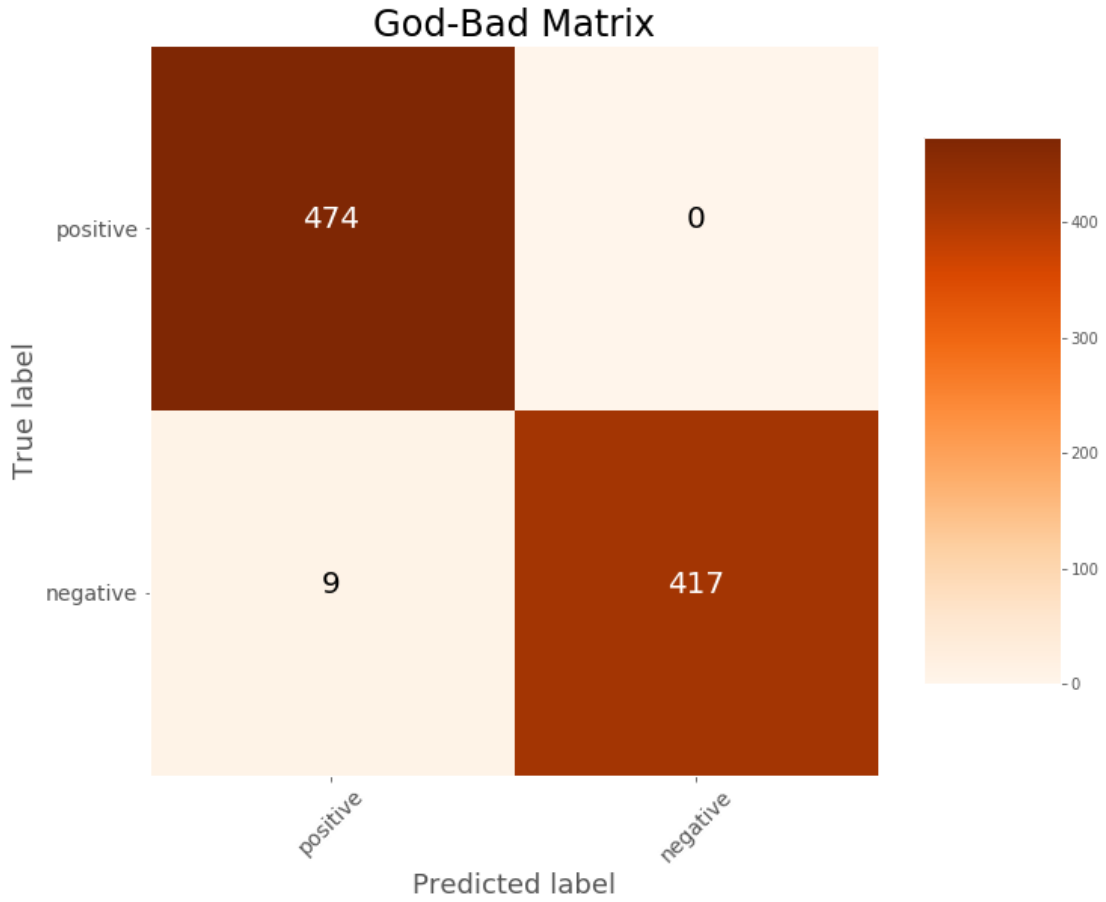

Рисунок 22 – Матрица результатов (тестовые данные)

Тренировать модель, которая точно предсказывает результаты, - это замечательно, но в большинстве случаев нам нужны не просто предсказания, мы хотите иметь возможность интерпретировать свою модель.

Важные функции доступны не только для линейных моделей. Большинство реализаций произвольного леса (RF) также обеспечивают показатели важности функций. Фактически, техника важности атрибутов, которую мы здесь представим (важность перестановок с опущенным столбцом), применима к любой модели, хотя, похоже, это понимают лишь немногие специалисты по машинному обучению. Важность перестановок с опущенным столбцом - это обычный, достаточно эффективный и очень надежный метод. Он напрямую измеряет важность переменной, наблюдая влияние на точность модели случайного тасования каждой переменной предиктора. Этот метод широко применим, потому что он не зависит от внутренних параметров модели, таких как коэффициенты линейной регрессии.

Я рекомендуем использовать важность перестановок для всех моделей, включая линейные модели, потому что вы можете в значительной степени избежать любых проблем с интерпретацией параметров модели.

Вывод:

Итог работы наглядно продемонстрирован на модели «Random forest», что работа с данными обработанными и проанализированными выдаёт меньшую погрешность, нежели необработанные данные (0.4% против 3.1%). Да, погрешность с необработанными данными не так уж сильно увеличилась, для наглядности, по данным ЦБ в 2018 году было выдано кредитов более чем на 3 триллиона рублей. 0,4% от этой суммы составляет 12 миллиардов рублей, а 3,1% от предоставленной суммы составляет 93 миллиарда рублей. Таким образом, можно было бы сэкономить ЦБ РФ 81 миллиард рублей. С такими цифрами прирост погрешности в 2,7% не кажется таким уж и маленьким.

В данном примере данных было не так много, и они были достаточно корректными, что даже без обработки выдали малую погрешность, но в рабочих реалиях люди сталкиваются с куда большим объёмом «грязных» данных, что приводит к большим погрешностям, так что при работе с данными, не поленитесь, сделайте анализ, чистку и обработку данных, это для вашего же, дальнейшего удобства.

5. ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ

Целью данной главы ВКР является разработка алгоритма для анализа и предобработки данных.

Достижение такой цели обеспечено выполнением следующих задач:

4.1.1 Разработать алгоритм для предобработки данных

4.1.2Реализовать методику предобработки данных с помощью Python;

4.1.3Проверить правильность реализации, созданной методики;

4.1.4Произвести сравнительный анализ результатов работы различных реализаций.

Разрабатываемая информационная система предназначена для использования в финансовых сферах. Пользователями программы выступают Аналитики данных, дата инженеры, отвечающие за процессы сбора, подготовки и очистки данных.

Чтобы перейти к рассмотрению потенциальных потребителей необходимо установить целевой рынок и провести его сегментирование.

5.1 Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения

5.1.1 Анализ конкурентных решений

В качестве конкурентных технических решений были выбраны следующие разработки:

1. Разработка алгоритма для предобработки и анализа данных на платформе Python (данная работа) (1);
2. Разработка алгоритма для предобработки и анализа данных на платформе C# (2);
3. Разработка алгоритма для предобработки и анализа данных на платформе Java (3).

Анализа выполнялся с помощью оценочной карты. Результаты

конкурентного анализа приведены в Таблице 2.

Таблица 2 – Оценочная карта

Критерии оценки		Вес крите-рия	Баллы			Конкуренто-способность		
			$Б_1$	$Б_2$	$Б_3$	$К_1$	$К_2$	$К_3$
1		2	3	4	5	6	7	8
Технические критерии оценки ресурсоэффективности								
1.	Скорость работы	0,3	5	4	5	1,5	1,2	1,5
2.	Гибкость платформы	0,2	5	3	5	1	0,6	1
3.	Простота эксплуатации	0,1	5	4	4	0,5	0,4	0,4
4.	Потребность в ресурсах	0,1	4	4	5	0,4	0,4	0,5
5.	Функциональные возможности	0,1	5	4	4	0,5	0,4	0,4
Экономические критерии оценки эффективности								
6.	Предполагаемый срок эксплуатации	0,05	5	4	4	0,25	0,2	0,2
7.	Обслуживание программного обеспечения	0,05	5	3	3	0,25	0,15	0,15
8.	Цена	0,1	5	5	5	0,5	0,5	0,5
Итого:		**1**	**39**	**31**	**35**	**4,9**	**3,85**	**4,65**

Конкурентоспособность решения 1 к решению 2 равна 1,27.

Конкурентоспособность решения 1 к решению 3 равна 1,05.

На основе вышеприведенных оценочной карты и значений конкурентоспособности можно сделать вывод, что решение 1 имеет конкурентные преимущества и является наиболее практичной для реализации. Именно такое решение и использовалось при выполнении текущего проекта.

5.1.2 SWOT-анализ

SWOT-анализ является одним из самых часто используемых методов в менеджменте и маркетинге. Он позволяет комплексно рассмотреть состояние момента на текущий момент и позволяет достичь понимания того, какие действия должны быть предприняты для нейтрализации слабых сторон и угроз, а также для максимизирования возможностей на основе сильных сторон проекта. Результаты SWOT-анализа представлены в Таблице 3.

Таблица 3 – SWOT-анализ

	Сильные стороны проекта: С1. Быстрота действия, разработанного ПО. С2. Графическое отображение результатов. С3. Универсальный алгоритм с мин. погрешностью. С4. Гибкость платформы проекта.	**Слабые стороны проекта:** Сл1. Необходимо наличие доступа ксети Internet для получения файлов наборов данных. Сл2. Необходимость ручного задания некоторого числа параметров. Сл3. Большое число этапов перед получением результатов. Сл4. Строго заданная структура файлов наборов данных.
Возможности: В1. Рост спроса на подобные разработки. В2. Рост рыночной стоимости подобных разработок. В3. Малое число конкурентных разработок. В4. Рост спроса на аналитические статьи, которые можно генерировать исходя из результатов приложения.	**В1В2С2С3** Разработка интерфейса и общий доступ к документации к разработке упростит интеграцию новыми потребителями. **В3В4С2С3** Проведение активной маркетинговой кампании с целью популяризации программы в финансовых сферах услуг и не только.	**В1В2В4Сл2Сл3** Автоматизация определения параметров анализа, повышение уровня интерактивности процесса.
Угрозы: У1. Сбои в работе сети Internet. У2. Изменение структуры файлов наборов данных. У3. Появление типов анализов данных, имеющих преимущество над используемым. У4. Появление конкурентного продукта.	**У1С2С3С4** Введение логов процесса работы ПО. **У2С1С3** Обеспечение текстового вывода результатов. **У3С3С4** Усовершенствование методики анализа. **У4С4** Отслеживание рыночной ситуации с целью внедрения актуальных и востребованных функций данного рода разработок	**У1Сл1Сл2Сл4** Обеспечение поддержки локального выбора файла набора данных. **У2Сл2Сл4** Избежание прямолинейности и строгого задания извлекаемой структуры получаемых данных.

Таким образом, в результате SWOT-анализа были выявлены направления, по которым необходимо развивать проект. Наиболее важными задачами на краткосрочный период являются усовершенствование методики анализа данных и создания интерфейса программного обеспечения для упрощения взаимодействия с пользователем.

5.2 Определение возможных альтернатив проведения научных исследований

В качестве морфологических характеристик объекта исследования можно выделить тип данных, язык программирования, тип анализа данных и тип приложения. Морфологическая матрица с рассмотрением альтернативных решений приведена в Таблице 4.

Таблица 4 – Морфологическая матрица альтернативных решений

Альтернативы	**1**	**2**	**3**
А. Тип данных	CSV	Grib	Grib2
Б. Язык программирования	Python	C#	Java
В. Тип анализа данных	Корреляционный	Статистический	На основе искусственных нейронных сетей
Г. Тип приложения	Веб-клиент	Десктопное приложение	Мобильное приложение

Данную работу можно описать как выбор наиболее желательных функционально конкретных решений вида А1Б1В3Г1:

1. тип данных: CSV – это простой формат файла, используемый для хранения табличных данных, таких как электронная таблица или база данных. Файлы в формате CSV можно импортировать и экспортировать из программ, которые хранят данные в таблицах, таких как Microsoft Excel или OpenOffice Calc. CSV означает «значения, разделенные запятыми».

2. язык программирования: Python – универсален, кроссплатформенен, имеет крупную базу готовых решений, библиотек и инструкций;

3. тип анализа данных: Нейронные сети - исключительно мощный метод моделирования, позволяющий воспроизводить чрезвычайно сложные зависимости.;

4. тип приложение: Веб-клиент – клиентское программное обеспечение, представляющее собой браузер и использующее http/https протоколы. Приложение не требует инсталляцию или загрузку программных модулей на рабочую станцию пользователя. Допускается установка дополнительных

общесистемных библиотек и использование защищенных сетевых протоколов, и от этого десктопным приложением не становится.

5.3 Планирование научно-исследовательских работ

5.3.1 Структура работ в рамках научного исследования

Перечень этапов и работ в рамках научного исследования представлен в Таблице 5.

Таблица 5 – Перечень этапов, работ, распределение исполнителей

Основной этап	№	Содержание работ	Исполнитель
Разработка ТЗ	1	Разработка требований	Научный руководитель (100%)
	2	Анализ требований	Научный руководитель (25%), инженер-программист (75%)
	3	Анализ предметной области	Научный руководитель (50%), инженер-программист (50%)
	4	Выбор инструментов и методологий	Научный руководитель (50%), инженер-программист (50%)
Проектирование ПО	5	Проектирование метода по обработке данных	Инженер-программист (100%)
	7	Проектирование компонент и классов ПО	Инженер-программист (100%)
Разработка ПО	8	Создание логики доступа к CSV-файлам	Инженер-программист (100%)
	9	Разработка и создание алгоритма	Инженер-программист (100%)
	10	Создание нейронных сетей	Инженер-программист (100%)
Тестирование и отладка	11	Тестирование ПО с различными данными	Инженер-программист (100%)
Обобщение и оценка	12	Оценка полученных результатов работы	Научный руководитель (75%), инженер-программист (25%)
результатов			
Оформление отчета по НИР	13	Оформление расчетно-пояснительной записки	Инженер-программист (100%)
Подведение итогов	14	Оценка выполненной работы	Научный руководитель (50%), инженер-программист (50%)

5.3.2 Определение трудоемкости выполнения работ

Необходимо произвести оценку трудоемкости выполнения вышеперечисленных работ. Для этого необходимо рассмотреть минимальное и максимальное время выполнения каждой из этих работ. Расчет ожидаемого значения трудоемкости будет выполняться по следующей формуле:

$$t_{ож,i} = \frac{(3t_{\min\ i} + 2t_{\max\ i})}{5} \quad (1)$$

где $t_{min\ i}$ – минимальное время выполнение i-ой работы чел.-дн.;

$t_{max\ i}$ – максимальное время выполнения i-ой работы чел.-дн.

Параллельность работ не учитывается, потому как инженер-программист является единственным исполнителем задач.

5.3.3 Разработка графика проведения научного исследования

Для подсчета продолжительности выполнения i-ой работы в календарных днях применяется формула:

$$T_{кi} = T_{рi} * k_{кал} \quad (2)$$

где $T_{рi}$ – продолжительность выполнения i-ой работы в рабочих днях;

$k_{кал}$ – коэффициент календарности.

Коэффициент календарности считается по формуле:

$$k_{кал} = \frac{T_{кал}}{T_{кал} - T_{вых} - T_{пр}} \quad (3)$$

где $T_{кал}$ –количество календарных дней в году;

$T_{вых}$ – количество выходных дней в году;

$T_{пр}$ – количество праздничных дней в году.

$$k_{кал\ 2018} = \frac{365}{365 - 118} = 1{,}48 \quad (4)$$

Временные показатели проведения научного исследования приведены в

Таблице 6.

Таблица 6 – Временные показатели научного исследования

Этап	**Исполнители**	**Продолжительность работ, дни**			**Длительность работ, чел/дн.**			
					T_{pi}		**T_{ki}**	
		t_{min}	**t_{max}**	**$t_{ож}$**	**НР**	**ИП**	**НР**	**ИП**
Разработка требований	НР	1	2	1,4	1,4	-	2	-
Анализ требований	НР, ИП	1	2	1,4	0,35	1,05	1	2
Анализ предметной области	НР, ИП	3	5	3,8	1,9	1,9	3	3
Выбор инструментов и методологий	НР, ИП	5	10	7	3,5	3,5	6	6
Создание нейронных сетей	ИП	2	10	5,2	-	5,2	-	8
Проектирование компонент и классов ПО	ИП	4	8	5,6	-	5,6	-	9
Создание логики доступа к CSV-файлам	ИП	10	15	12	-	12	-	18
Проектирование метода по обработке данных	ИП	15	20	17	-	17	-	26
Разработка и создание алгоритма	ИП	10	15	12	-	12	-	18
Тестирование ПО с различными данными	ИП	5	10	7	-	7	-	11
Оценка полученных результатов работы	НР, ИП	2	2	2	1,5	0,5	3	1
Оформление расчетно-пояснительной записки	ИП	15	20	17	-	17	-	26
Оценка выполненной работы	НР, ИП	5	10	7	3,5	3,5	6	6
Итого:				98,4	12,15	86,25	21	134

НР – Научный руководитель; ИП – Инженер-программист.

На основе приведенной таблицы строится календарный план-график. В таком плане-графике предусмотрены разбиения по декадам (10 дней) и месяцам.

Таблица 7 – Календарный план-график работ

№	Вид работ	Испол-нители	T_{ki}, кал.-дн.		Продолжительность выполнения работ													
					январь		февраль			март			апрель			май		
			НР	ИП	2	3	1	2	3	1	2	3	1	2	3	1	2	3
1	Разработка требований	НР	2	-														
2	Анализ требований	НР, ИП	1	2														
3	Анализ предметной области	НР, ИП	3	3														
4	Выбор инструментов и методологий	НР, ИП	6	6														
5	Разработка и создание алгоритма	ИП	-	8														
7	Проектирование компонент и классов ПО	ИП	-	9														
8	Создание логики доступа к CSV-файлам	ИП	-	18														
9	Проектирование метода по обработке данных	ИП	-	26														
10	Создание нейронных сетей	ИП	-	18														
11	Тестирование ПО с различными данными	ИП	-	11														
12	Оценка полученных результатов работы	НР, ИП	3	1														
13	Оформление расчетно-пояснительной записки	ИП	-	26														
14	Оценка выполненной работы	НР, ИП	6	6														

5.3.4 Бюджет научно-технического исследования (НТИ)

Расчет материальных затрат НТИ

Материальные затраты определяются материалами и ресурсами, используемыми в течении процесса разработки текущего научно-исследовательского проекта. К таким, в данном случае, можно отнести только электроэнергию, обеспечивающую работу ЭВМ инженера-программиста. Канцелярские принадлежности предусмотрены расходами научной организации.

В час компьютер потребляет около 0,3 кВт. Возьмем стоимость 1 кВт*ч электричества для юридических лиц равной 5,8 рублей. В день компьютер

используется в среднем 6 часов. Тогда, стоимость материальных затрат:

$$З_{мат} = 0{,}3 \text{ кВт} * 5.8 \frac{\text{руб}}{\text{кВт} * \text{ч}} * 6 \text{ ч} * 134 \text{ д} = 1398{,}96 \text{ руб.} \quad (5)$$

Расчет на специальное оборудование для научных - экспериментальных работ

Затраты на специальное оборудование будет состоять из монитора, системного блока, периферийного оборудования. В совокупности затраты выйдут в размере 11500 рублей.

Основная заработная плата исполнителей темы

Расчет бюджета НТИ состоит из расчета материальных затрат, затрат на зарплаты руководителю и инженера. Материальные затраты составляют только расходные материалы и амортизация оборудования.

Оклад руководителя от ТПУ (доцента, к.т.н) составляет 26300 рубля (без учета районного коэффициента).

Оклад младшего научного сотрудника составляет 17000 руб. (без учета районного коэффициента).

Таблица 8 – Распределение рабочего времени

Показатели рабочего времени	**Дни**
Календарное число	365
Количество нерабочих дней (праздники/выходные)	118
Действительный годовой фонд рабочего времени	247

С учетом районного коэффициента, равного 30% от оклада, получается месячная заработная плата:

$$З_м^{рук} = 26300 * 1{,}3 = 34190 \text{ руб.}$$
$$З_м^{разр} = 17000 * 1{,}3 = 22100 \text{ руб.} \quad (6)$$

Зная месячную заработную плату каждого участника проекта, можно рассчитать соответствующую среднедневную заработную плату. Количество месяцев работы без отпуска принимается равным 11,2 (считается отпуск длиною

24 рабочих дня при 6-дневной рабочей неделе):

$$З_{дн}^{рук} = \frac{34190 * 11,2}{247} = 1550,31 \text{ руб.}$$
$$З_{дн}^{разр} = \frac{22100 * 11,2}{247} = 1002,1 \text{ руб.} \quad (7)$$

Тогда, основная зарплата за период НТИ:

$$З_{осн}^{рук} = 1550,31 * 21 = 32556,51 \text{ руб.},$$
$$З_{осн}^{разр} = 1002,1 * 134 = 134292,12 \text{ руб.} \quad (8)$$

За статью расходов «Затраты по основной зарплате»:

$$З_{осн} = 32556,51 + 134292,12 = 166848,63 \text{ руб.} \quad (9)$$

Дополнительная заработная плата исполнителей темы

С учётом основной заработной платы, можно посчитать дополнительную заработную плату в размере 12 % от основной:

$$З_{доп}^{рук} = k_{доп} * З_{осн} = 0,12 * 32556,51 = 3906,78 \text{ руб.}$$
$$З_{доп}^{разр} = k_{доп} * З_{осн} = 0,12 * 134292,12 = 16115,05 \text{ руб.} \quad (10)$$

За статью расходов «Затраты по дополнительной зарплате»:

$$З_{доп} = 3906,78 + 16115,05 = 20021,83 \text{ руб.} \quad (11)$$

Отчисления во внебюджетные фонды (страховые отчисления)

Величина отчислений во внебюджетные фонды определяется как:

$$З_{внеб1} = k_{внеб}(З_{осн} + З_{доп}) = 0.28 * (32556,51 + 3906,78) = 11011,91$$
$$З_{внеб1} = k_{внеб}(З_{осн} + З_{доп}) = 0.28 * (134292,12 + 16115,05) = 45422,97 \quad (12)$$

За статью расходов «Отчисления во внебюджетные фонды»:

$$З_{внеб} = 11011,91 + 45422,97 = 56434,88 \text{ руб.} \quad (13)$$

Расчет затрат на научные и производственные командировки

Научных и производственных командировок в данном объекте исследования не производилось.

Контрагентные расходы

Данная статья расходов описывает затраты, связанные с привлечением сторонних организаций для выполнения работ связанных с текущим объектом исследования. Привлечения сторонних организаций не производилось, потому контрагентные расходы равны нулю.

Накладные расходы

Накладные расходы учитывают прочие затраты организации. Коэффициент учета накладных расходов берется равным 0,16. Тогда, расчет накладных расходов:

$$З_{накл} = (\text{сумма статей } 1 - 7) * 0{,}16 = 40992{,}68 \text{ руб.} \quad (14)$$

Формирование бюджета затрат научно-исследовательского проекта

Таблица 9 – Бюджет затрат по каждому исполнению НТИ

Наименование статьи	Сумма руб.		
	Исп. 1	Исп. 2	Исп. 3
1. Материальные затраты НТИ	1398,96	1357,20	1461,60
2. Затраты на спец. оборудование	11500		
3. Затраты по основной з/п	166848,63	170581,06	171300,20
4. Затраты по доп. з/п	20021,83	20469,72	20556,02
5. Отчисления во внебюджетные фонды	56434,88	57697,36	57940,58
6. Затраты на научные и производственные командировки	0		
7. Контрагентские расходы	0		
8. Накладные расходы	40992,68	41856,85	42041,34
9. Бюджет затрат НТИ	297196,98	303462,19	304799,74

5.4 Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования

Определение эффективности происходит посредством нахождения двух средневзвешенных величин: финансовой эффективности и ресурсоэффективности.

Интегральный финансовый показатель разработки

рассчитывается как:

$$I_{\text{финр}}^{\text{исп.}i} = \frac{\Phi_{\text{р},i}}{\Phi_{max}}, \qquad (15)$$

Где $\Phi_{\text{р},i}$ – стоимость *i*-го варианта исполнения;

Φ_{max} - максимальная стоимость исполнения научно- исследовательского проекта.

Тогда, согласно Таблице 10:

$$I_{\text{финр}}^{\text{исп1}} = \frac{297196{,}98}{304799{,}74} = 0{,}975, \qquad (16)$$

$$I_{\text{финр}}^{\text{исп2}} = \frac{303462{,}19}{304799.74} = 0{,}996, \qquad (17)$$

$$I_{\text{финр}}^{\text{исп3}} = 1$$

Интегральный показатель ресурсоэффективности можно определить следующим образом:

$$I_{\text{р},i} = \sum a_i b_i \qquad (18)$$

где $I_{\text{р,i}}$ – интегральный показатель ресурсоэффективности для i-го варианта разработки;

a_i – весовой коэффициент i-го варианта разработки;

b_i – бальная оценка i-го варианта исполнения разработки, устанавливаемая экспертным путем по выбранной шкале оценивания.

Расчет интегральных показателей ресурсоэффективности приведен в Таблице 10.

Таблица 10 – Сравнительная оценка характеристик вариантов исполнения проекта.

Критерии оценки	Вес критерия	Баллы		
		Исп. 1	Исп. 2	Исп. 2
1. Скорость работы	0,3	5	4	5
2. Гибкость платформы	0,25	5	3	5
3. Простота эксплуатации	0,15	5	4	4
4. Потребность в ресурсах	0,1	4	4	5
5. Функциональные возможности	0,2	5	4	4
Итого:	1	4,9	3,75	4,65

$$I_{\text{р}-\text{исп}1} = 0{,}3 * 5 + 0{,}25 * 5 + 0{,}15 * 5 + 0{,}1 * 4 + 0{,}2 * 5 = 4{,}9;$$
$$I_{\text{р}-\text{исп}2} = 0{,}3 * 4 + 0{,}25 * 3 + 0{,}15 * 4 + 0{,}1 * 4 + 0{,}2 * 4 = 3{,}75; \quad (19)$$
$$I_{\text{р}-\text{исп}3} = 0{,}3 * 5 + 0{,}25 * 5 + 0{,}15 * 4 + 0{,}1 * 5 + 0{,}2 * 4 = 4{,}65.$$

Интегральный показатель эффективности вариантов исполнения разработки рассчитывается на основании интегрального показателя ресурсоэффективности и интегрального финансового показателя согласно формуле:

$$I_{\text{исп}.i} = \frac{I_{\text{р}-\text{исп}i}}{I_{\text{финр}}^{\text{исп}.i}}. \quad (20)$$

Сравнительная эффективность проекта считается согласно формуле:

$$\text{Э}_{\text{ср}} = \frac{I_{\text{исп}.1}}{I_{\text{исп}.2}}. \quad (21)$$

Расчет сравнительной эффективности разработок приведет в Таблице 11.

Таблица 11. Сравнительная эффективность разработки

Показатели	**Исп.1**	**Исп.2**	**Исп.3**
Интегральный финансовый показатель разработки	0,975	0,996	1
Интегральный показатель ресурсоэффективности разработки	4,9	3,75	4,65
Интегральный показатель эффективности	5,026	3,765	4,65
Сравнительная эффективность вариантов исполнения	1,335	1	1,235

Вывод по разделу: Исходя из проведенного анализа сравнительной эффективности вариантов исполнения можно сделать вывод, что исполнение 1 является более эффективным и предпочтительным вариантом из всех предложенных. Так же, согласно подсчету бюджета затрат по каждому из вариантов следует подчеркнуть, что Исполнение 1 является так же самым выгодным. Таким образом, в настоящей работе реализована самая подходящая вариация проекта.

6 СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ

Введение

В данной работе будут проанализированы полученные данные, сделана предварительная обработка и чистка данных, сравнение результатов, обработанных данных и «грязных» данных, а также проведён их скрининг.

Анализ и обработка данных велась исключительно при помощи компьютера. В данном разделе будут рассмотрены вопросы выполнения требований к безопасности труда, к промышленной безопасности, охране окружающей среды и ресурсосбережению, непосредственно связанные с работой с ПКВ стандарте ГОСТ 12.0.003–2015 «Опасные и вредные производственные факторы» рассматриваются вредные и опасные факторы, подразделяющиеся по природе действия на следующие группы [11]:

6.1 Производственная безопасность

- Физические;
- Химические;
- Биологические;
- Психофизиологические.

Так как разработка приложения ведется в кабинете при использовании персонального компьютера (далее ПК), выделяют следующие вредные и опасные факторы (Таблица 12).

Таблица 12– Опасные и вредные факторы при выполнении работ

Факторы *(по ГОСТ 12.0.003-2015)*		Нормативные документы
Вредные	Опасные	
1. Отклонение показателей микроклимата; 2. Недостаточная освещенность рабочей зоны; 3.Повышенный уровней Шума и вибрации от Вентиляторов охлаждения компьютера; 4. Умственное Перенапряжение и монотонность труда.	1. Удары электрическим током; 2. Короткое замыкание; 3. Статическое электричество.	1. ГОСТ 12.0.003–2015; 2. СанПин 2.2.4–548–96 [11]; 3. СанПиН 2.2.2/2.4.1340–03 [13]; 4. СанПиН 2.2.1/2.1.1.1278–03 [14]; 5. ГОСТ Р 50739-95 [15].

6.1.2 Анализ выявленных вредных факторов проектируемой среды

Показатели микроклимата должны обеспечивать сохранение теплового баланса человека с окружающей средой и поддержание оптимального или допустимого теплового состояния организма. Показателями, характеризующими микроклимат в помещении, являются:

- Температура воздуха;
- Температура поверхностей;
- Относительная влажность воздуха;
- Скорость движения воздуха;

Оптимальные показатели температур воздуха, поверхностей, относительной влажности и скорости воздуха должны соответствовать значениям, указанным в таблице 13 и распространяться на всю рабочую зону. Температура воздуха в рабочей зоне в течение смены не должна выходить за пределы оптимальных величин, указанных таблице 13 для отдельных категорий работ. Выполняемая работа относится к категории Iа –

производимая сидя и сопровождающаяся незначительным физическим напряжением. Перепады температуры воздуха по высоте и по горизонтали, а также изменения температуры воздуха в течение смены, при обеспечении оптимальных величин микроклимата на рабочих местах, не должны превышать 2 °С и выходить за пределы величин, указанных в таблице 13.

Таблица 13 – Параметры микроклимата для помещений, где установлен компьютер согласно СанПиН 2.2.4–548–96

Время года	Параметр микроклимата			
	Температура воздуха, °С	Температура поверхностей	Влажность, %	Скорость движения воздух, м/с
Холодный	22-24	21-25	40-60	0,1
Теплый	23-25	22-26	40-60	0,1

В целях профилактики неблагоприятного воздействия микроклимата должны быть использованы защитные мероприятия, такие как системы местного кондиционирования воздуха, компенсация неблагоприятного воздействия одного параметра микроклимата изменением другого, помещения для отдыха и обогревания, регламентация времени работы, в частности, перерывы в работе, сокращение рабочего дня.

6.1.3 Расчет уровня шума

Согласно СанПиН 2.2.2/2.4.1340–03 в производственных помещениях с использованием ПК уровни шума на рабочих местах не должны превышать предельно допустимых значений. Шум на рабочем месте вызван следующем оборудованием: винчестером в системном блоке, вентиляторами, кулерами охлаждения процессора ПК, монитор, клавиатура. Уровень шума на рабочем месте не должен превышать 50 дБ. В таблице 14 приведены уровни шума из различных источников.

Таблица 14 – Уровень звукового давления различных источников

Источник шума	Уровень шума, дБ
Жесткий диск	40
Вентилятор	45
Монитор	17
Клавиатура	10
Принтер	45
Сканер	42

Уровень шума, возникающий от нескольких некогерентных источников, работающих одновременно, подсчитывается на основании принципа энергетического суммирования излучений отдельных источников.

Подставив в формулу значения уровня звукового давления для каждого оборудования, получим:

$10* lo (10^{4}+10^{4.5}+10^{1.7}+10^{1})$ 46,2 дБ

Полученное значение не превышает допустимую норму, поэтому использование специальных средств защиты не требуется. В случае превышения допустимой нормы для снижения уровня шума стены и потолок помещений, где установлены компьютеры, могут быть оснащены звукопоглощающими материалами.

6.1.4 Освещенность

Производственное освещение — это система устройств и мер, обеспечивающих благоприятную работу зрения человека в процессе труда. Правильно спроектированное и выполненное производственное освещение улучшает условия зрительной работы, снижает утомляемость, способствует повышению производительности труда, благотворно влияет на производственную среду, оказывая положительное психологическое воздействие на работника, повышает безопасность труда и снижает травматизм. Недостаточность освещения приводит к напряжению зрения, ослабляет внимание, приводит к наступлению преждевременной утомленности. Чрезмерно яркое освещение вызывает ослепление, раздражение и резь в глазах.

Неправильное направление света на рабочем месте может создавать резкие тени, блики, дезориентировать работающего.

К системам производственного освещения предъявляются следующие требования:

- Соответствие уровня освещённости рабочих мест характеру выполняемой зрительной работы;
- Достаточно равномерное распределение яркости на рабочих поверхностях и в окружающем пространстве;
- отсутствие резких теней, прямой и отраженной повышенной яркости светящихся поверхностей, вызывающей ослепление;
- постоянство освещённости во времени;
- оптимальная направленность излучаемого осветительными приборами светового потока;
- долговечность, экономичность, пожаро и электробезопасность, эстетичность, удобство и простота эксплуатации.

Согласно СанПиН 2.2.2/2.4.1340-03, рабочие столы следует размещать таким образом, чтобы мониторы были ориентированы боковой стороной к световым проемам, чтобы естественный свет падал преимущественно слева, искусственное освещение в помещениях должно осуществляться системой общего равномерного освещения.

Естественное освещение осуществляется через два оконных проема размером 2 на 2,5 метра по наружной стене. Нормируемые показатели естественного, искусственного и совмещенного освещения в соответствии с СанПиНом 2.2.1/2.1.1.1278–03 указаны в таблице 15.

Таблица 15– Нормируемые показатели естественного, искусственного и совмещенного освещения в соответствии с СанПиНом 2.2.1/2.1.1.1278–03

Помещения	Рабочая поверхность и плоскость нормирования КЕО и освещенности и высота плоскости над полом, м	Естественное освещение КЕО ен, %		Совмещенное освещение КЕО ен, %		Искусственное освещение			
						освещенность, лк		показатель дискомфорта, М, не более	коэффициент пульсации освещённости, Кп, %, не более
		при верхнем или комбинированном освещении	при боковом освещении	при верхнем или комбинированном освещении	при боковом освещении	при комбинированном освещении	при общем освещении		
1	2	3	4	5	6	7	8	9	10
Кабинеты	Г- 0,8	3,2	1,0	1,8	06	400	300	40	15

Чтобы поддерживать освещение в помещении по всем соответствующим нормам, необходимо хотя бы два раза в год проводить чистку стекал и светильников, а также по мере необходимости заменять перегоревшие лампы. В утреннее и вечернее время вводится общее искусственное освещение. Основными источниками искусственного освещения являются люминесцентные лампы белого и дневного света ЛБ-20 и ЛД-20. Следует ограничивать отраженную блескость на рабочих поверхностях (экран, стол, клавиатура) за счет правильного выбора и расположения светильников, яркость бликов на экране не должна превышать 40 кд/м^2. Рассматриваемое помещение соответствует указанным нормированным показателям, как в дневное время суток, так и в вечернее.

6.1.5 Монотонность труда и умственное перенапряжение

Монотонный режим работы связан с однообразным повторением рабочих операций. Опасность монотонности заключается в снижении внимания к процессу производства, быстрой утомляемости и снижению интереса к трудовому процессу, что может повлиять на безопасность труда в целом. Для борьбы с монотонностью используются следующие меры:

- расширение круга обязанностей, усложнение работы или объединение ее в комплексы;

- организация 5-ти минутных перерывов;
- увеличение числа поставленных целей, за счет разделения одной общей на несколько промежуточных.

Умственное перенапряжение возникает в результате длительной умственной работе и проявляется в снижении работоспособности. Факторами, вызывающими умственное перенапряжение, являются:

- Длительное и повышенное внимание и концентрация;
- Высокие требования к самореализации и самоподготовке;
- Ощущение недостатка времени
- Неправильное питание работника (приводит к недостатку энергии для работы мозга).

Для борьбы с умственным перенапряжением используется выполнение различных физических упражнений, а также организация 5-ти минутных перерывов и перерыва на обед.

6.1.6 Техника электробезопасности

В соответствии с СанПиНом 2.2.2/2.4.1340–03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы» помещения, где размещаются рабочие места с ПК, должны быть оборудованы защитным заземлением в соответствии с техническими требованиями по эксплуатации электроустановок и вычислительной техники. Рабочие места с ПК не следует размещать вблизи силовых кабелей и вводов, высоковольтных трансформаторов, технологического оборудования, создающего помехи в работе ПК.

Помещение, в котором проводилась работа, относится к помещениям без повышенной опасности, так как в нем отсутствуют условия, которые могут создать повышенную или особую опасность.

К организационным мерам электробезопасности относятся:

1. Инструктаж

Целью инструктажа является сообщение работникам знаний,

необходимых для правильного и безопасного выполнения ими своих профессиональных обязанностей, а также формирования у работников убеждения в объективной и абсолютной необходимости выполнения правил и норм безопасной жизнедеятельности в производственной среде.

Существуют следующие виды инструктажа:

5.1.3.1 вводный инструктаж

5.1.3.2 первичный инструктаж

5.1.3.3 периодический (повторный)

2. Правильная организация рабочего места

Организация рабочего места заключается в выполнении ряда мероприятий, которые обеспечивают рациональный и безопасный трудовой процесс, и эффективное использование орудий и предметов труда, что повышает производительность и способствует снижению утомляемости работающих. Так, например, правильно выбранная рабочая поза (с возможностью её перемены) исключает или сводит к минимуму вредное влияние выполняемой работы на организм человека.

3. Режим труда и отдыха

Оптимальный режим труда и отдыха – это такое чередование периодов работы с периодами отдыха, при котором достигается наибольшая эффективность деятельности человека и хорошее состояние его здоровья. Он оказывает благотворное влияние на функциональное состояние человека.

Электрические установки, источником работы которых является переменный ток напряжением 220В и частота 50 Гц, к которым относится большинство оборудования ПК, представляют для человека большую потенциальную опасность, так как в процессе эксплуатации (проведение регламентных работ) человек может коснуться частей оборудования, находящихся под напряжением. Специфическая опасность электроустановок состоит в том, что токоведущие проводники, оказавшиеся под напряжением в результате повреждения изоляции, не подают каких-либо сигналов, которые бы предупреждали об опасности. Для защиты от поражения электрическим током

все токоведущие части должны быть защищены от случайных прикосновений и заземлены. Питание устройства должно осуществляться от силового щита через автоматический предохранитель, срабатывающий при коротком замыкании нагрузки.

В соответствии с правилами электробезопасности в помещении осуществляется постоянный контроль состояния электропроводки, предохранительных щитов, шнуров, с помощью которых включаются в электросеть ПК, осветительные приборы, другие электроприборы. Также в помещении отсутствуют токопроводящая пыль, электрически активная среда, возможность одновременного прикосновения к металлическим частям прибора и заземляющему устройству, высокая температура и сырость.

Основным опасным фактором является опасность поражения электрическим током. Исходя из анализа состояния помещения, данное помещение по степени опасности поражения электрическим током можно отнести к классу помещений без повышенной опасности. В помещении подавляющая часть электрической проводки является скрытой. Поражение электрическим током возможно только при возникновении оголенных участков на кабеле, а также нарушении изоляции распределительных устройств, однако в помещении кабель имеет двойную изоляцию, поэтому опасность поражения значительно снижается. Не исключается также опасность поражения и от токоведущих частей компьютера в случае их пробоя и нарушении изоляции.

Для устранения опасности поражения электрическим током регулярно проводится осмотр кабелей, проводов, электрических розеток и токоведущих частей компьютера. А также, перед началом работы за компьютером каждый работник проходит инструктаж по технике безопасности.

Возникающие при прикосновении к любому из элементов ПК разрядные токи статического электричества могут привести к выходу ПК из строя. Для снижения величины возникающих зарядов статического электричества в помещении покрытие полов выполнено из однослойного поливинилхлоридного антистатического линолеума. К мерам защиты от статического электричества

также можно отнести общее и местное увлажнение воздуха.

Компьютер также является и источником статического электричества. Местами скопления статических зарядов, как правило, служит поверхность экрана монитора. Для уменьшения статического электричества на поверхности монитора следует раз в 6 часов протирать экран влажной материей.

6.2 Экологическая безопасность

В работе по реализации проекта не оказывается значительного влияния на окружающую среду, так как в процессе работы не использовались вредные химические соединения. Самой серьезной проблемой, с которой столкнулись при выполнении работы, является потребление электроэнергии. С увеличением количества компьютерных систем, внедряемых в производственную сферу, увеличится и объем потребляемой ими электроэнергии. Рост энергопотребления приводит к таким экологическим нарушениям, как изменение климата – накопление углекислого газа в атмосфере Земли (парниковый эффект).

Из этого можно сделать простой вывод, что необходимо стремиться к снижению энергопотребления, то есть разрабатывать и внедрять системы с малым энергопотреблением. В современных компьютерах, повсеместно используются режимы с пониженным потреблением электроэнергии при длительном простое.

Техника, вышедшая из строя, утилизируется согласно ГОСТ Р 50739–95[15]. Люминесцентные лампы, вышедшие из строя, сдаются в специализированный пункт приема. Для уменьшения отходов, связанных с расходными материалами (бумага, ручки, картриджи и т.д.), можно использовать повторно переработанную бумагу или использовать двухстороннюю печать.

6.3 Безопасность в чрезвычайных ситуациях

Чрезвычайная ситуация - это обстановка на определенной территории, сложившаяся в результате аварии, опасного природного явления, катастрофы, стихийного или иного бедствия, которые могут повлечь или повлекли за собой человеческие жертвы, ущерб здоровью людей или окружающей природной

среде, значительные материальные потери и нарушение условий жизнедеятельности людей.

Мероприятия по обеспечению пожарной безопасности

Пожарная безопасность регламентируется федеральным законом

«Технический регламент о требованиях пожарной безопасности». Опасными факторам пожара для людей являются открытый огонь, искры, повышенная температура воздуха и предметов, токсичные продукты горения, дым, пониженная концентрация кислорода, обрушение и повреждение зданий, сооружений, установок, а также взрыв.

Организационными мероприятиями по обеспечению пожарной безопасности являются: обучение рабочих и служащих правилам пожарной безопасности; разработка и реализация норм и правил пожарной безопасности, инструкций о порядке работы в помещениях; изготовление и применение средств наглядной агитации по обеспечению пожарной безопасности.

Основной причиной возникновения пожара в помещениях с электронной техникой является неисправность проводки. Вероятность возгорания самих электронных устройств чрезвычайно мала.

В качестве оперативных средств тушения пожара применяются порошковые огнетушители ОПУ–5. Сеть электропитания оборудуется входным рубильником, позволяющим в оперативном порядке отключить электропитание во всем здании. Для обеспечения эвакуации людей в случае пожара помещения должны иметь не менее двух выходов шириной не менее одного метра и высотой не менее двух метров.

Наиболее частыми причинами пожаров являются нарушения правил пожарной безопасности и технологических процессов, неправильная эксплуатация электросети и оборудования, грозовые разряды.

Одна из главных причин травм, связанных с действием электрического тока, слабые знания правил электробезопасности. Нарушение правил электробезопасности при использовании электроустановок и непосредственное

соприкосновение с токоведущими частями электроустановок, находящихся под напряжением, создает опасность поражения электрическим током.

6.4 Правовые и организационные вопросы обеспечения безопасности

Под безопасностью понимаются защитные мероприятия и средства, обеспечивающие снижение опасности до минимальной степени риска, когда негативные факторы не превышают допустимой величины.

Трудовой кодекс регулирует вопросы, относящиеся к охране и организации труда. СанПиН 2.2.2/2.4.1340–03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы» регламентирует аспекты работы на ПК. Так же ГОСТ 12.2.032–78[16] ССБТ «Рабочее место при выполнении работ сидя» регламентирует общие требования к рабочему месту при выполнении работ сидя.

Эргономика рабочего места

Данные правила определяют санитарно – эпидемиологические требования к: проектированию, изготовлению и эксплуатации ПК, используемых на производстве; организации рабочих мест с ПК, производственным оборудованием. Согласно СанПиН 2.2.2/2.4.1340–03 Общие требования к организации рабочих мест пользователей ПК:

- При размещении рабочих мест с ПК расстояние между рабочими столами должно быть не менее 2,0 м, а расстояние между боковыми поверхностями видеомониторов не менее 1,2 м.
- Рабочие места с ПК в помещениях с источниками вредных производственных факторов должны размещаться с организованным воздухообменом.
- Рабочее место сотрудника, требующее значительного умственного напряжения или высокой концентрации внимания, рекомендуется изолировать руг от друга перегородками высотой 1,5–2,0 м.
- Конструкция рабочего кресла должна обеспечивать поддержание

рациональной рабочей позы при работе на ПК позволять изменять позу с целью снижения напряжения мышц шейно – плечевой области и спины для предупреждения развития утомления.

- Площадь на одно рабочее место с компьютером для взрослых пользователей должна составлять не менее 6 $м^2$, а объем не менее -24 $м^3$.
- Помещения с компьютерами должны оборудоваться системами отопления, кондиционирования воздуха или эффективной приточно-вытяжной вентиляцией.
- Для внутренней отделки интерьера помещений с компьютерами должны использоваться диффузно-отражающие материалы с коэффициентом отражения для потолка – 0,7–0,8; для стен – 0,5–0,6; для пола – 0,3–0,5.
- Поверхность пола в помещениях эксплуатации компьютеров должна быть ровной, без выбоин, нескользкой, удобной для очистки и влажной уборки, обладать антистатическими свойствами.
- В помещении должны находиться аптечка первой медицинской помощи, а также углекислотный огнетушитель для тушения пожара.

Вывод по разделу. При выполнении раздела социальная ответственность были установлены и исследованы на соответствие нормам вредные и опасные факторы. По результатам исследований были рассмотрены меры по противодействию данным факторам. Так же была рассмотрена экологическая безопасность, где было установлено, как следует утилизировать отходы при выполнении работы. В разделе ЧС были рассмотрены возможные виды ЧС и меры их профилактики. И в последнем разделе были рассмотрены правовые и организационные моменты социальной ответственности. Рабочее место соответствуют нормативным требованиям.

Заключение

a. Разработали алгоритм для предобработки данных;

- Проверка пустых значений в столбцах(missing’s)
- Проверить на наличие дубликатов строк
- Проверить на резко отличающихся значения от среднего показателя в столбце(выбросы)
- Проверить на колинеарность
- Привести данные к единому формату (нормализация)

b. Реализовали методику предобработки данных с помощью Python;

Смотреть приложение (А)

c. Произвели сравнение над обработанными и необработанными данными;

Выявили, что работа с обработанными данными дает меньший процент погрешности, чем работа с необработанными данными.

d. Произвели сравнительный анализ результатов работы различных реализаций;

Выявили, что механизм обучения данных (перестановок с опущенным столбцом) является наиболее эффективным, хоть и более производительным.

Список используемых источников

1. Скрининг – один из способов работы банка с заемщиком [Электронный ресурс]. URL http://www.incred.ru/pub/skrining-i-svedeniya-o-nem-dlya-klienta/51757/, свободный. – Яз. Рус. Дата обращения 01.05.2019г.
2. Сборка Caffe в Google Colaboratory: бесплатная видеокарта в облаке [Электронный ресурс]. URL https://habr.com/ru/post/413229/ свободный. – Яз. Рус. Дата обращения 01.05.2019г.
3. Python // Python Software Foundation. URL: https://www.python.org/ (дата обращения: 15.05.2018).
4. NumPy // NumPy . URL: http://www.numpy.org/ (дата обращения: 15.05.2018)
5. Хочу все знать. Язык SAS [Электронный ресурс]. URL https://geekbrains.ru/posts/sas_lang свободный. – Яз. Рус. Дата обращения 20.05.2019г
6. Data Mining Fruitful and Fun // Orange. URL: https://orange.biolab.si/ (дата обращения: 15.05.2018)
7. The Comprehensive R Archive Network // R Project for Statistical Computing. URL: https://cran.r-project.org/ (дата обращения: 15.05.2018)
8. Елисеева И. И., Юзбашев М. М. Общая теория статистики. М.: Финансы и Статистика, 2002. – 480 с.
9. Beware Default Random Forest Importances [Электронный ресурс]. URL https://explained.ai/rf-importance/, свободный. – Яз. Рус. Дата обращения 06.05.2019г.
10. Специализированные массивы // Всероссийский научно исследовательский институт гидрометеорологической информации - Мировой центр данных. URL: http://meteo.ru/data (дата обращения: 15.05.2018)
11. ГОСТ 12.0.003-2015. ССБТ. «Опасные и вредные производственные факторы. Классификация».
12. СанПиН 2.2.4.548–96. «Гигиенические требования к

микроклимату производственных помещений».

13. СанПиН 2.2.2/2.4.1340–03. «Санитарно-эпидемиологические правила и нормативы «Гигиенические требования к персональным электронно- вычислительным машинам и организации работы».

14. СанПиН 2.2.1/2.1.1.1278–03. «Гигиенические требования к естественному, искусственному и совмещённому освещению жилых и общественных зданий».

15. ГОСТ Р 50739–95. «Средства вычислительной техники. Защита от несанкционированного доступа к информации».

16. ГОСТ 12.2.032-78 ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требовании. – М.: Изд-во стандартов.

Приложение А

```
import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

plt.style.use('ggplot')

data = pd.read_csv('./crx
(2).csv',header=None,na_values='?',error_bad_lines=False,
delimiter=',')

data.shape

data.tail()

data.head()

data.columns = ['A' + str(i) for i in range(1, 16)] + ['class']

data['A5'][5]

categorical_columns = [c for c in data.columns if
data[c].dtype.name == 'object']

numerical_columns   = [c for c in data.columns if
data[c].dtype.name != 'object']

categorical_columns, numerical_columns

data.describe()

data[categorical_columns].describe()

for c in categorical_columns:

    print (data[c].unique())

from pandas.tools.plotting import scatter_matrix

scatter_matrix(data, alpha=0.05, figsize=(10, 10));

print (data.corr())

data['A1'].describe()

data = data.fillna(data.median(axis=0), axis=0)

data.count(axis=0)

data = data.fillna(data.median(axis=0), axis=0)

data['A1'].describe()

data['A1'] = data['A1'].fillna('b')

data_describe = data.describe(include=[object])

for c in categorical_columns:

    data[c] = data[c].fillna(data_describe[c]['top'])
```

```
print(data.describe(include=[object]))

binary_columns    = [c for c in categorical_columns if
data_describe[c]['unique'] == 2]

nonbinary_columns = [c for c in categorical_columns if
data_describe[c]['unique'] > 2]

print (binary_columns, nonbinary_columns)

data.at[data['A1'] == 'b', 'A1'] = 0

data.at[data['A1'] == 'a', 'A1'] = 1

print(data['A1'].describe())

data_describe = data.describe(include=[object])

print(data_describe)

for c in binary_columns[1:]:

    top = data_describe[c]['top']

    top_items = data[c] == top

    data.loc[top_items, c] = 0

    data.loc[np.logical_not(top_items), c] = 1

print(data[binary_columns].describe())

print(data['A4'].unique())

data_nonbinary = pd.get_dummies(data[nonbinary_columns])

print (data_nonbinary.columns)

data_numerical = data[numerical_columns]

data_numerical = (data_numerical - data_numerical.mean()) /
data_numerical.std()

print(data_numerical.describe())

data = pd.concat((data_numerical, data[binary_columns],
data_nonbinary), axis=1)

data = pd.DataFrame(data, dtype=float)

print (data.shape)

print (data.columns)

X = data.drop(('class'), axis=1)  # Выбрасываем столбец 'class'.

y = data['class']

feature_names = X.columns

print (feature_names)

print (X.shape)

print (y.shape)
```

```
N, d = X.shape

from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size = 0.3, random_state = 11)

print (binary_columns, nonbinary_columns)

data['A1'].describe()

from sklearn import ensemble

from sklearn import preprocessing

rf = ensemble.RandomForestClassifier(n_estimators=4,
random_state=11)

lab_enc = preprocessing.LabelEncoder()

encoded = lab_enc.fit_transform(y_train)

print(rf.fit(X_train, encoded))


err_train = np.mean(encoded != rf.predict(X_train))

#err_test  = np.mean(y_test  != rf.predict(X_test))

err_test = np.mean(lab_enc.fit_transform(y_test) !=
rf.predict(X_test))

print (err_train, err_test)

importances = rf.feature_importances_

indices = np.argsort(importances)[::-1]


print("Feature importances:")

for f, idx in enumerate(indices):

    print("{:2d}. feature '{:5s}' ({:.4f})".format(f + 1,
feature_names[idx], importances[idx]))

d_first = 20

plt.figure(figsize=(8, 8))

plt.title("Feature importances")

plt.bar(range(d_first), importances[indices[:d_first]],
align='center')

plt.xticks(range(d_first),
np.array(feature_names)[indices[:d_first]], rotation=90)

plt.xlim([-1, d_first]);

best_features = indices[:8]

best_features_names = feature_names[best_features]
```

```
print(best_features_names)
from sklearn.metrics import confusion_matrix
import itertools

def plot_confusion_matrix(cm, classes,
                          normalize=False,
                          title='Confusion matrix',
                          cmap=plt.cm.Oranges):
    if normalize:
        cm = cm.astype('float') / cm.sum(axis=1)[:, np.newaxis]
        print("Normalized confusion matrix")
    else:
        print('Confusion matrix, without normalization')

    print(cm)

    plt.figure(figsize = (10, 10))
    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    plt.title(title, size = 24)
    plt.colorbar(aspect=4)
    tick_marks = np.arange(len(classes))
    plt.xticks(tick_marks, classes, rotation=45, size = 14)
    plt.yticks(tick_marks, classes, size = 14)

    fmt = '.2f' if normalize else 'd'
    thresh = cm.max() / 2.

    # Labeling the plot
    for i, j in itertools.product(range(cm.shape[0]),
range(cm.shape[1])):
        plt.text(j, i, format(cm[i, j], fmt), fontsize = 20,
                 horizontalalignment="center",
                 color="white" if cm[i, j] > thresh else "black")
```

```
    plt.grid(None)
    plt.tight_layout()
    plt.ylabel('True label', size = 18)
    plt.xlabel('Predicted label', size = 18)
predict_result = rf.predict(X_train[:50])
print(y_train[:50])
print(predict_result)
cm = confusion_matrix(lab_enc.fit_transform(y_train[:3000]),
rf.predict(X_train[:3000]))
plot_confusion_matrix(cm, classes = ['positive', 'negative'],
                      title = 'God-Bad Matrix')
from sklearn.ensemble.forest import _generate_unsampled_indices


def oob_classifier_accuracy(rf, X_train, y_train):
    X = X_train.values
    y = y_train.values


    n_samples = len(X)
    n_classes = len(np.unique(y))
    predictions = np.zeros((n_samples, n_classes))
    for tree in rf.estimators_:
      unsampled_indices =
_generate_unsampled_indices(tree.random_state, n_samples)
      tree_preds = tree.predict_proba(X[unsampled_indices, :])
      predictions[unsampled_indices] += tree_preds


    predicted_class_indexes = np.argmax(predictions, axis=1)
    predicted_classes = [rf.classes_[i] for i in
predicted_class_indexes]


    oob_score = np.mean(y == predicted_classes)
    return oob_score
def permutation_importances(rf, X_train, y_train, metric):
  baseline = metric(rf, X_train, y_train)
```

```
  imp = []
  for col in X_train.columns:
    save = X_train[col].copy()
    X_train[col] = np.random.permutation(X_train[col])
    m = metric(rf, X_train, y_train)
    X_train[col] = save
    imp.append(baseline - m)
  return np.array(imp)
imp = permutation_importances(rf, X_train, y_train,
oob_classifier_accuracy)
indices = np.argsort(imp)[::-1]


print("Feature imp:")
for f, idx in enumerate(indices):
    print("{:2d}. feature '{:5s}' ({:.4f})".format(f + 1,
feature_names[idx], imp[idx]))
d_first = 20
plt.figure(figsize=(8, 8))
plt.title("Feature imp")
plt.bar(range(d_first), imp[indices[:d_first]], align='center')
plt.xticks(range(d_first),
np.array(feature_names)[indices[:d_first]], rotation=90)
plt.xlim([-1, d_first]);
best_features = indices[:8]
best_features_names = feature_names[best_features]
print(best_features_names)
from sklearn.metrics import confusion_matrix
import itertools

def plot_confusion_matrix(cm, classes,
                          normalize=False,
                          title='Confusion matrix',
                          cmap=plt.cm.Oranges):
    if normalize:
```

```
        cm = cm.astype('float') / cm.sum(axis=1)[:, np.newaxis]
        print("Normalized confusion matrix")
    else:
        print('Confusion matrix, without normalization')

    print(cm)

    plt.figure(figsize = (10, 10))
    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    plt.title(title, size = 24)
    plt.colorbar(aspect=4)
    tick_marks = np.arange(len(classes))
    plt.xticks(tick_marks, classes, rotation=45, size = 14)
    plt.yticks(tick_marks, classes, size = 14)

    fmt = '.2f' if normalize else 'd'
    thresh = cm.max() / 2.

    # Labeling the plot
    for i, j in itertools.product(range(cm.shape[0]),
range(cm.shape[1])):
        plt.text(j, i, format(cm[i, j], fmt), fontsize = 20,
                 horizontalalignment="center",
                 color="white" if cm[i, j] > thresh else "black")

    plt.grid(None)
    plt.tight_layout()
    plt.ylabel('True label', size = 18)
    plt.xlabel('Predicted label', size = 18)
cm = confusion_matrix(lab_enc.fit_transform(y_test[:3000]),
rf.predict(X_test[:3000]))
plot_confusion_matrix(cm, classes = ['positive', 'negative'],
                      title = 'God-Bad Matrix')
```